TU MICRO

COMMODORE

N.º 1 · SEGUNDA EPOCA · 350 PTAS (IVA INCLUIDO)

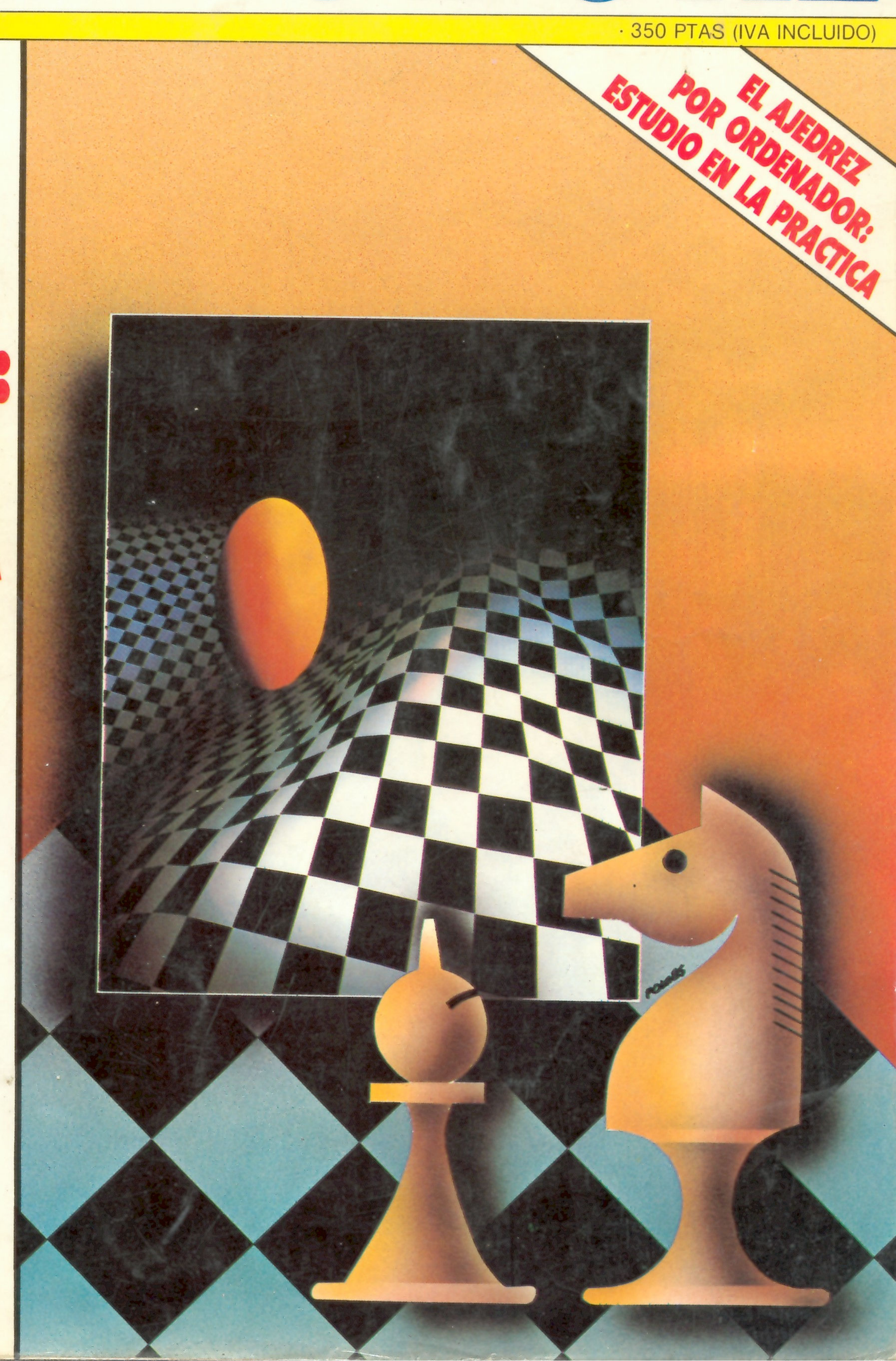
EL AJEDREZ POR ORDENADOR: ESTUDIO EN LA PRACTICA

ROBOTICA PARA C-64

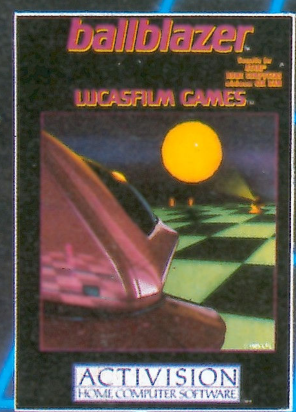
JET-MAN: MAGNIFICO JUEGO TECLA A TECLA

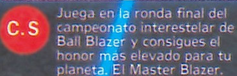
TABLA DE MNEMONICOS DEL 6502

EASY SCRIPT: PROCESO DE TEXTOS PROFESIONAL

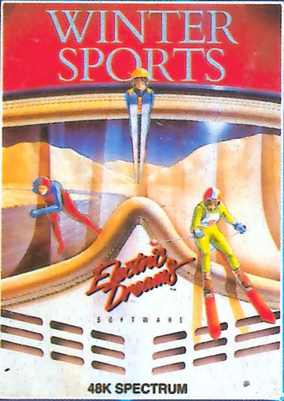


ACTIVISION
HOME COMPUTER SOFTWARE





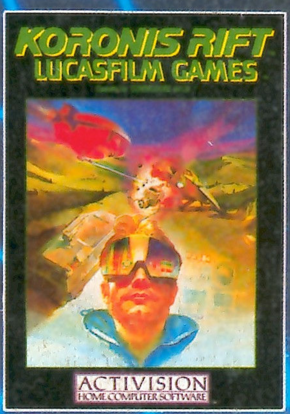
C.S Juega en la ronda final del campeonato interestelar de Ball Blazer y consigues el honor mas elevado para tu planeta. El Master Blazer.



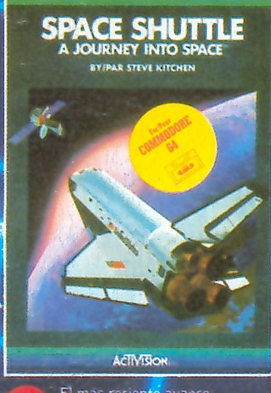
S. A Si no puedes practicar tus deportes de invierno favoritos te ofrecemos oportunidad de hacerlo con tu ordenador. Ocho competiciones distintas te esperan. Apresurate a equiparte y disfruta con el descenso, slalom, salto ski etc.



CSA Ya puedes disfrutar en tu ordenador con la pelicula que nos ofrece Steven Spielberg. No esperes ir al cine. Incluye la banda sonora original.



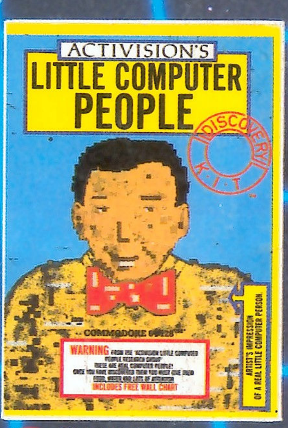
C Encuentra un tesoro en un remoto sistema estelar, pero ¡cuidado! con sus guardianes. Si quieres sobrevivir elabora un cuidadoso plan. La recompensa esta fuera de este mundo.



C.S. El mas reciente avance tecnico. Vd. puede realizar una jornada completa dentro de una capsula espacial desde que se levanta de la superficie de la tierra y acude a un encuentro en el espacio hasta que aterriza nuevamente. Compruebe su habilidad.



C Descubre los secretos de Eidolon a traves de una misteriosa maquina del siglo XIX, y sumergete en un mundo subterraneo poblado de curiosas criaturas magicas y feroces dragones guardianes.



C Ya hemos descubierto a unos pequeños seres en el ordenador, ahora tienes que cuidar de no contradecirlos y que esten lo mas contentos posible. ¡Intentalo!



S El laberinto magico. Introducete en el movimiento tridimensional del planeta Newgama III y encuentra las piezas necesarias para completar el robot.

... y sus clásicos GHOSTBUSTER. C.S.A.M. DECATHLON DE ACTIVISION. C RIVER RAID. C.S.M.

✻ ENDURO S PITFALL 2 C.S.M. y otros títulos más • SPACE SHUTTLE. C DISIGNER PENCIL. C ...etc.

ENCUENTRALO
EN LA DIVISION Online *DE* GALERIAS



Disponibles para:
COMMODORE C
SPECTRUM S
AMSTRAD A
MSX M

DELEGACION EN CATALUÑA:
C/Paris, 132 - 134
08011 Barcelona
Teléfono: 93-322 72 11

EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A:

PROEIN, S. A.
Velázquez, 10 - 28001 Madrid
Tels. 91/276 22 08/09

Director:
ANTONIO M. FERRER ABELLO

Redactor Jefe:
FERNANDO LOPEZ MARTINEZ

Redacción:
ANTONIO CARVAJAL
JUAN M. LOPEZ MARTINEZ
PABLO GARCIA MOLINA
IGNACIO BARCO LUENGO
ALFREDO SINDIN VALERO

Colaboradores:
JOSE LUIS VAZQUEZ DE PARGA
ANTONIO MANZANERA

Secretaria de Redacción:
PILAR MANZANERA AMARO

Diseño y Maquetación:
CARLOS GONZALEZ AMEZUA

Ilustraciones:
ANTONIO PERERA
RAMON POLO

Fotografía:
EQUIPO GALATA

Directora Publicidad:
CARMINA FERRER
Tel. (91) 457 69 23

Publicidad Barcelona:
ISIDRO IGLESIAS
Avda. Corts Catalanes, 1010
Tel. (93) 307 11 13

Director de Producción:
VICENTE ROBLES

Directora de Administración:
MARIA ANTONIA BUITRAGO

Suscripciones:
MARIA GONZALEZ AMEZUA

Redacción, administración, publicidad y suscripciones
Plza. República Ecuador, 2.
28016 MADRID. Tel. 250 58 20
Télex 49371 ELOC E

Dirección para correspondencia:
Aptdo. de Correos 61.294
28080 MADRID
TU MICRO COMMODORE es una publicación mensual de Ediciones INGELEK.

Fotocomposición:
ESPACIO Y PUNTO, S. A.

Fotomecánica:
RODACOLOR, S. A.

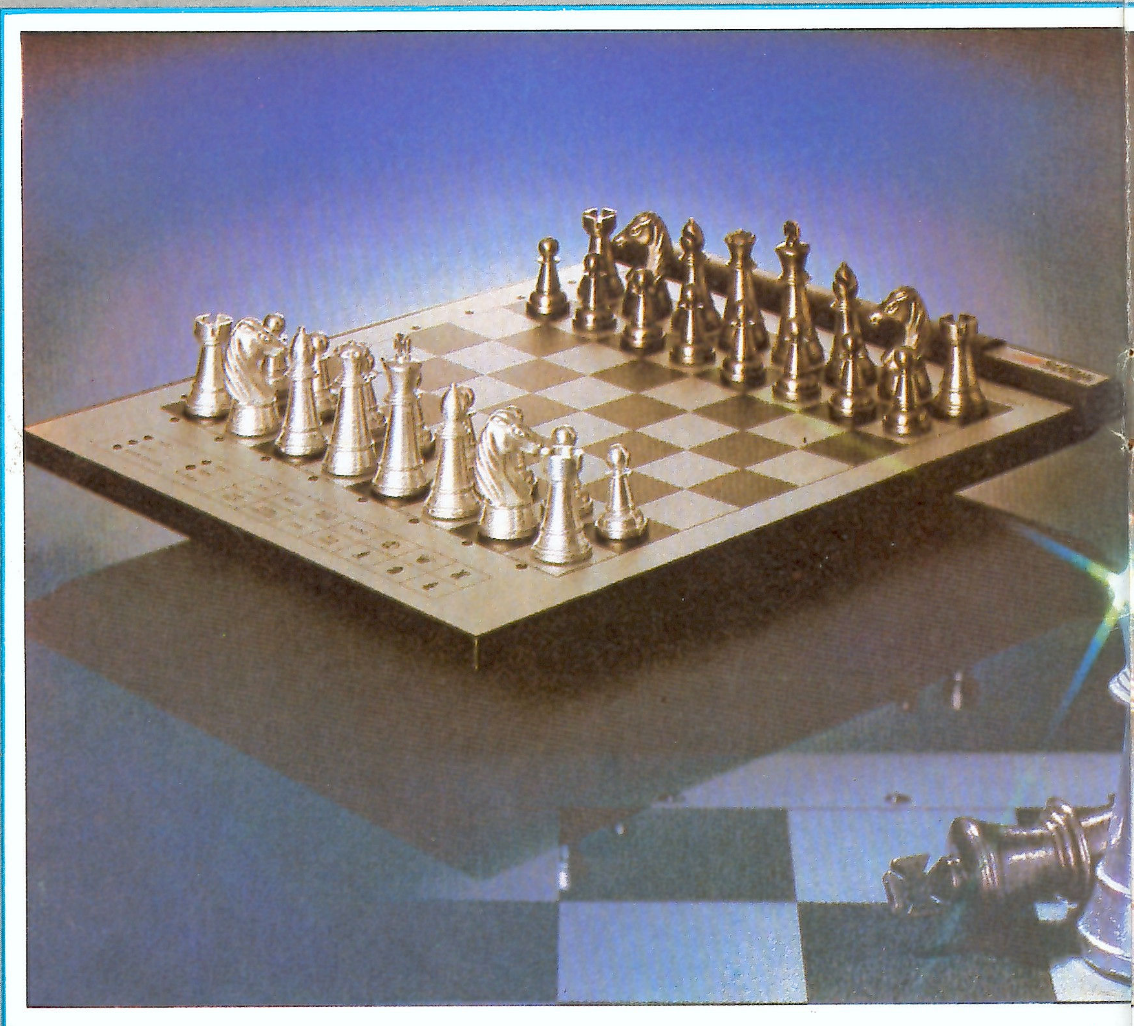
Impresión:
GRAFICAS REUNIDAS, S. A.

Distribución:
COEDIS, S. A.
Valencia, 245. Tel. 215 70 97
08007 BARCELONA
Almacén: Nacional II. Km. 609,4
MOLINS DE REI (Barcelona)
Delegación en Madrid: Serrano, 165.
Tel. 411 11 48
Almacén: Laforja, 19-21, esq. Hierro
Pol. Industrial Loeches
TORREJON DE ARDOZ (Madrid)

Precios para España. Ejemplar: 350 ptas. (IVA incluido). 330 ptas. (Canarias, Ceuta y Melilla).
(La suscripción anual incluye 11 números).
Distribución Cono Sur:
CADE, S.R.L.
Pasaje Sud América, 1532
Tel. 21 24 64
Buenos Aires 1.290. Argentina.

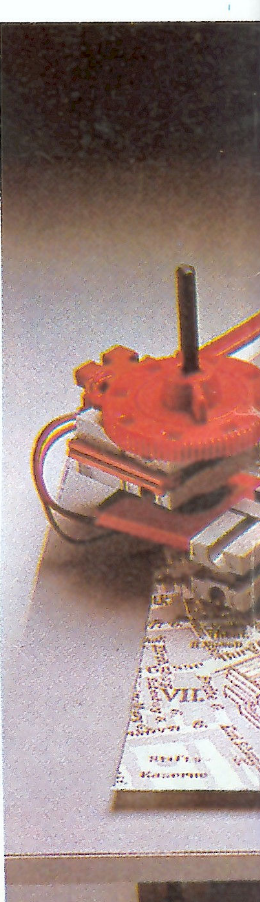
D. L.: M. 16.046-1986
Impreso en España.

TU MICRO COMMODORE N.° 1
Mayo 1986





Lo último y lo mejor en programas comerciales: Slicon Warrior, Punchy, Kung-Fu, Back to the Future y, como no, una vuelta ciclista: Tour de France.

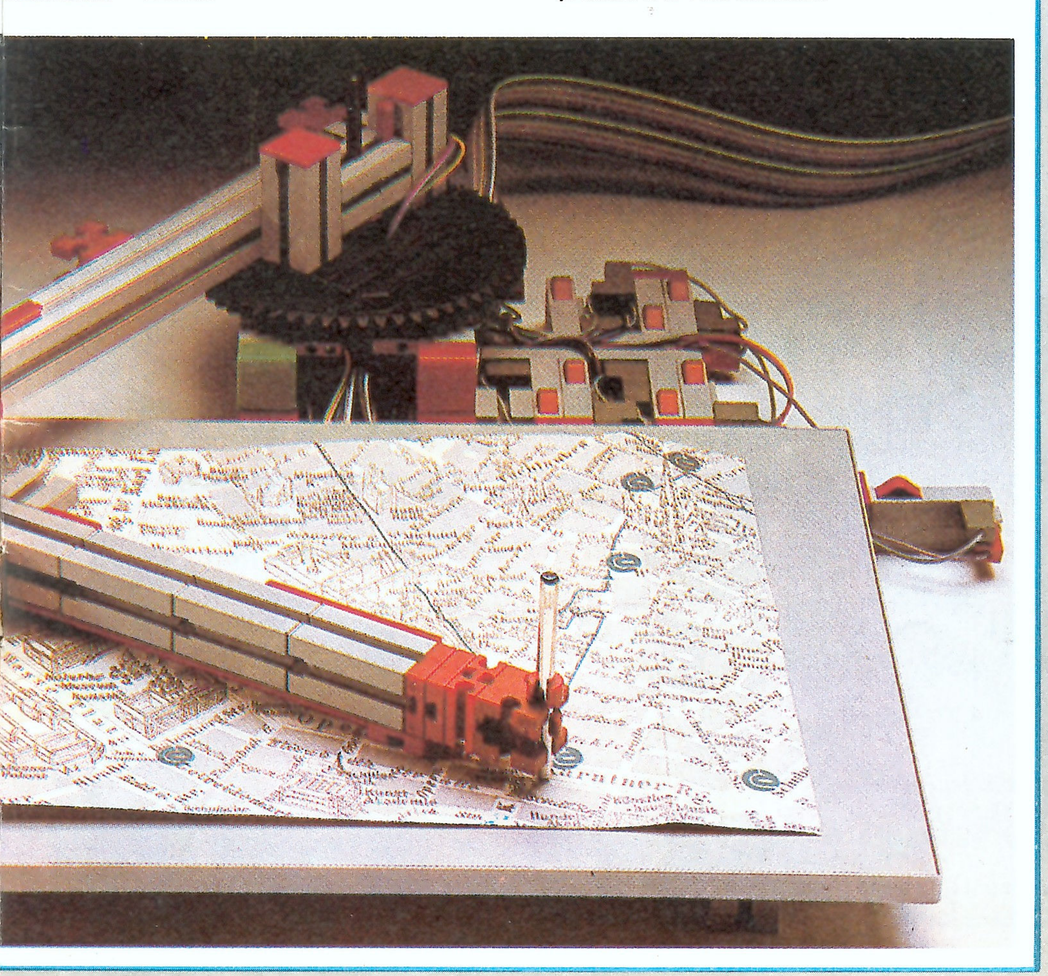




Los problemas de un astronauta con propulsión JET intentando recuperar las pilas de su transbordador de materia.

El comentario más amplio para el juego de los juegos: el ajedrez. La diversión en 64 casillas.

La robótica. Un antiguo argumento de ciencia ficción que ya se encuentra extraordinariamente próximo a la vida cotidiana.



Lo prometido es deuda, y como ya anunciamos el mes pasado, aquí nos encontramos de nuevo, en el kiosko, con una perioricidad mensual, y lo que es más importante con un nuevo... novísimo, TU MICRO COMMODORE. Confiamos en recibir próntamente vuestras opiniones sobre esta VUESTRA REVISTA.

NOTICIAS

Suscripción de la mano de los 7 Magníficos

El agraciado ganador de una suscripción por un año a nuestra revista, por su colaboración en la confección de Los 7 Magníficos del número 18, ha sido nuestro amigo de Almería:

Antonio Fco. Saa Quintas
El Almendro, 120
AGUADULCE - ALMERIA

Enhorabuena a ti Antonio, y ya sabéis, los demás animaos a enviarnos vuestros votos, os esperan muchas otras oportunidades de haceros con el premio.



Ganadores del segundo concurso de programación Tu Micro Commodore

Y ahora un notición. El jurado del segundo concurso de programación ha deliberado ya sobre los premiados en el citado certamen, emitiendo el siguiente veredicto:

1 PREMIO
Programa: Camino hacia el Oeste
Autor: Matilde de Murga

2 PREMIO
Programa: TAZ
Autor: Angel Navarro Batista

3 PREMIO
Programa: Esqui
Autor: Jordi Sais Riera

¡Enhorabuena a todos! Os rogamos que a la mayor brevedad os pongáis en contacto con nosotros para hacernos saber cuál es el material informático en el que deseáis invertir la dotación monetaria de vuestro premio.

Los galardonados podéis dirigiros a nosotros por carta a:

EDICIONES INGELEK
(TU MICRO COMMODORE)
PZA. REPUBLICA ECUADOR, 2 – 10
28016 MADRID

O bien efectuar un primer contacto telefónico en el 250 58 20 / 458 20 41, 457 69 23 o 250 55 79, anteponiendo el prefijo 91 si llamáis desde fuera de Madrid.

De feria va la cosa

Siempre se ha dicho que mayo es el mes de las flores, pero en este mundillo informático podemos asegurar sin temor a equivocarnos que mayo es el mes de la feria.

Versión II del Commodore PC

A partir del mes de abril será comercializado en España la versión II de los Commodore PC 10-II y PC 20-II, siendo fácilmente reconocibles las nuevas asignaciones ya que aparecen en el frontal del equipo.

Ambos modelos disponen de una memoria RAM de 512 Kbytes y están dotados de la placa supergráfica desarrollada especialmente para el Commodore AT. Este Adaptador Gráfico Avanzado (AGA) unifica las características de todas las placas gráficas existentes en el mercado, ofreciendo cuatro nuevos modos de trabajo, no soportados antes por primera placa.

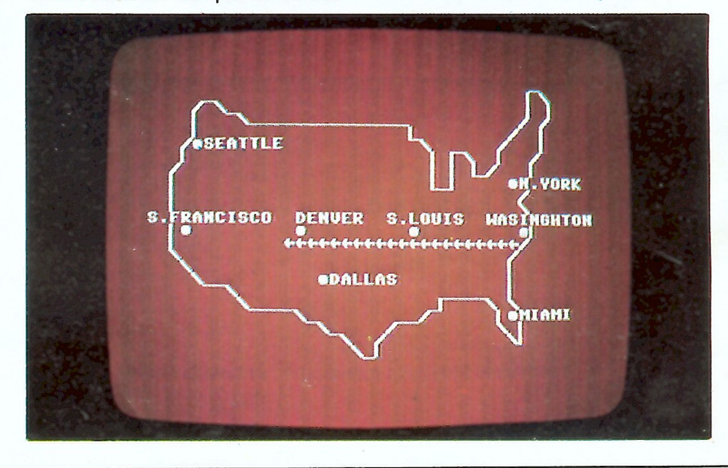
El AGA no requiere ninguna expansión del BIOS. La tecnología del AGA está basada en un sistema Gate Array, lo que permite que el tamaño de la placa sea de solamente 23 x 10 cm.

La nueva ROM de BIOS de nivel XT tiene ampliadas las rutinas de autotest. Los mensajes de error son ahora excepcionales comprensibles, y al poner en marcha el Sistema, la pantalla visualiza las direcciones de memoria donde se hallan los ports serie y paralelo.

Una de las nuevas características a reseñar en el modelo PC 20 es que la capacidad del disco duro es ahora de 20 Mbytes y se caracteriza por el poco calor que produce.

Internamente, la serie II ha sido optimizada: las dos placas principales de antes se han sustituido por una nueva más compacta. El teclado y el Sistema Operativo se entregan en su versión castellana.

Y la parte más grata de toda la noticia es que el incremento de las características de esta nueva serie no suponen un aumento en los precios y seguirán siendo de 353.000 ptas. para el PC 10-II y 520.000 ptas. para el PC 20-II.



A partir del próximo martes 13 (vaya con la fechita) y hasta el domingo 18, se desarrollará en Barcelona la bienal INFORMAT 86, que se constituye como una feria internacional de informática, en la cual toman parte las más destacadas firmas comerciales, dedicadas desde la informática de alto nivel hasta los microordenadores domésticos, así como las actividades relacionadas con el sector, tales como las casas de software o la prensa técnica.

Así pues, como ibamos a faltar nosotros. Podréis encontrarnos en el stand INGELEK, S.A., tanto en INFORMAT como en EXPOTRONICA, la feria internacional de electrónica que tendrá lugar en el mismo Recinto Ferial barcelonés. No olvidéis hacernos una visita, para los que no os podéis desplazar hasta la bella ciudad portuaria, publicaremos en nuestro próximo número de junio cumplida información sobre nuestras actividades en la exposición.

Soft Express entre en el mercado del software español

El pasado mes hizo su aparición en el mercado del software la firma SOFT EXPRESS, con las últimas novedades en el mercado para Commodore como Uridium y el cartucho Games Killer entre otros.

Los que estéis interesados en recibir más información dirigiros a:

SOFT EXPRESS
Duque Fernán Nuñez, 2, 4.°
Madrid

o a los teléfonos: 228 68 13 y 228 66 34 (prefijo 91 para fuera de Madrid).



Exclusiva mundial

Master-Hard presenta en España como primicia mundial el Convertidor de Monitores, mediante el cual cualquier tipo de monitor (incluido el sistema americano) en color o fósforo verde, se puede transformar en un completo y perfecto receptor de televisión o terminal de video. El sistema incorpora un selector de canales y una conexión para aparatos de vídeo.

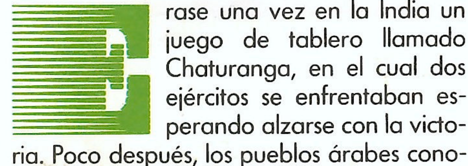
Con esta sencilla instalación, podemos ver en el monitor de nuestro ordenador, un programa de marcianitos como nuestra película favorita o un partido de fútbol, con la mayor nitidez que poseen los monitores.

El convertidor de monitores está comercializado y patentado a nivel mundial por MASTER-HARD. Magdalena, 213. FERROL – LA CORUÑA.



Ocho x Ocho

Arte, ciencia, juego, deporte o excusa para que seamos considerados deportistas sin necesidad de levantarnos del asiento, el ajedrez ha sido definido de muchas formas distintas. Durante los últimos años de la historia de este universal pasatiempo, muchos técnicos de la informática emplearon sus habilidades tratando de encontrar un programa capaz de vencer a cualquier ajedrecista. Presentamos aquí una muestra de los mejores programas para los ordenadores Commodore, auténticos maestros del tablero.



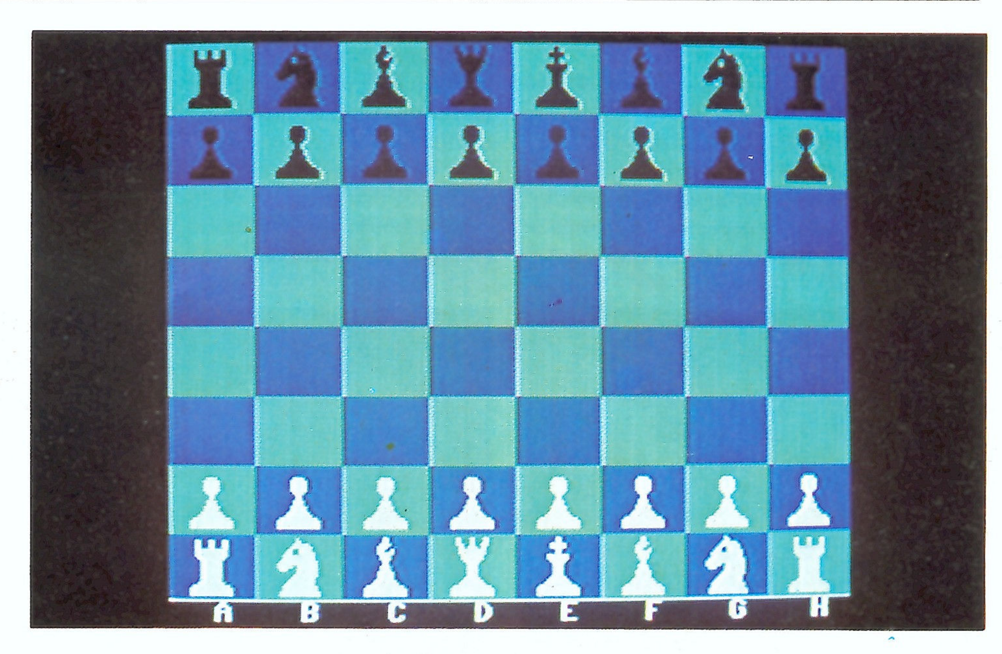
Erase una vez en la India un juego de tablero llamado Chaturanga, en el cual dos ejércitos se enfrentaban esperando alzarse con la victoria. Poco después, los pueblos árabes conocieron este curioso pasatiempo y dejaron de cultivar las tierras del desierto (por eso ahora el desierto está tan desprovisto de vegetación) para dedicarse de lleno a él.

La cultura europea debe mucho a los árabes, y una de las deudas contraidas fue el ajedrez, el cual pasó al viejo continente gracias a las invasiones musulmanas. Desde entonces, monarcas y plebeyos aprendieron a jugar, a utilizar sus cerebros en aras del ajedrez, lo cual permitió que aproximadamente en el siglo XIX ya existiera un reglamento uni-

versal del mismo.

Se originaron campeonatos locales, provinciales, y por fin mundiales. Los grandes jugadores se enfrentaban unos a otros con denodado esfuerzo, e iban enseñando, a través de sus publicaciones y las partidas que jugaban, cuanto sabían. Célebres ajedrecistas fueron Staunton, Anderssen, Morphy, Lasker, Capablanca, Alekhine..., todos ellos obtuvieron resonantes victorias.

En la actualidad, miles de personas subsisten gracias al ajedrez, ya sea como entrenadores, como jugadores o como teóricos. Ya son muy pocas las sorpresas que nos reserva cada partida porque existen multitud de estudios y opiniones acerca de tan controvertido pasamiento, que goza de carácter deportivo. Aún así, el ajedrez en España sólo tendrá una verdadera difusión popular cuando Roman Toran y Arturo Pomar jueguen sus partidas en el estadio Bernabeu, con cien mil espectadores coreando a voz en grito sus nombres.



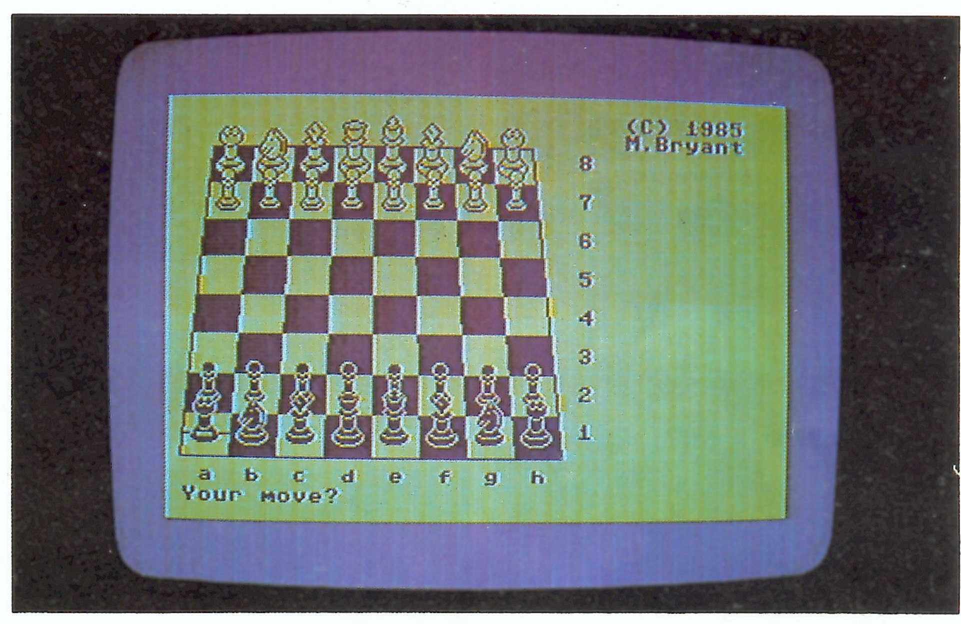


La familia Commodore 16, 64 y 128 Kbytes de ajedrecista

Realmente nadie se propuso crear los ordenadores para vencer al ajedrez, pero en vista de sus capacidades parecía algo obligado. Por ello, titánicos esfuerzos se han hecho, aunque sin obtener el rendimiento deseado. Verdaderas maravillas técnicas como el Mark VII, quizá lo más perfeccionado hasta el momento, son dominadas por los maestros del tablero.

La razón de este fracaso es evidente, pues los ordenadores no pueden albergar ideas estratégicas complejas en su cerebro de chips. A pesar de todo, dentro de las limitaciones impuestas por los circuitos de silicio, los logros conseguidos son brillantes. Tácticamente, son capaces de observar el reglamento a la perfección, realizar sacrificios (ceder piezas a cambio de algo, el jaque mate generalmente), etc. Por contra, en el campo de lo estratégico las deficiencias son mucho más numerosas, lo cual se observa con especial detenimiento en la fase final del juego.

Los commodorianos disponemos de varios programas preparados para jugar al ajedrez contra nuestros ordenadores. De entre todos ellos el equipo de investigación de software de TU MICRO COMMODORE ha seleccionado cuatro: Colossus Chess 4-0, Sargon II, Grandmaster 64E y Chess, basándose en la calidad y prestaciones que ofrecen los programas estudiados. Pasemos (¡por fin!) a conocer cada uno de ellos por separado.

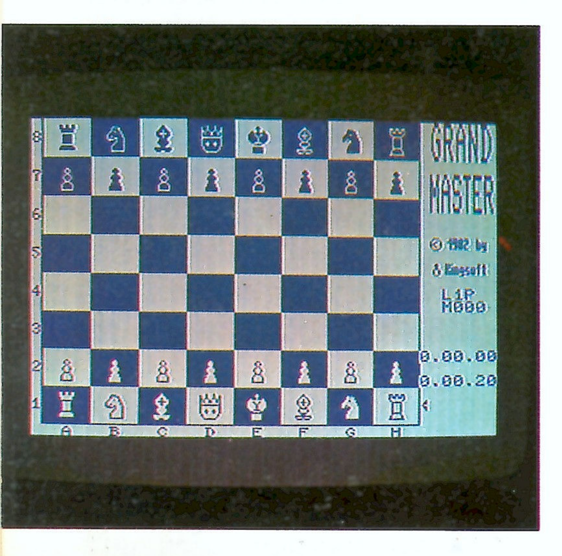




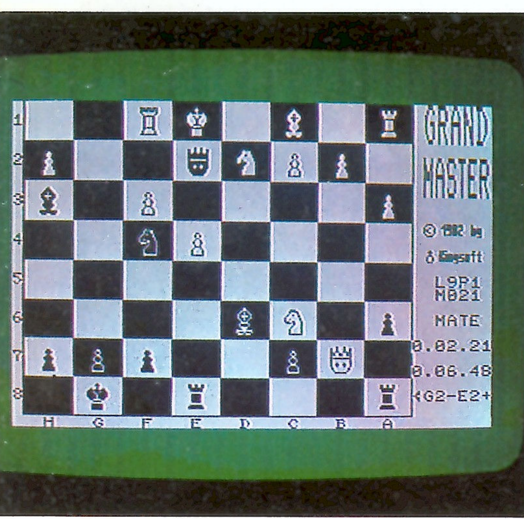
Colossus Chess 4-0. El número uno

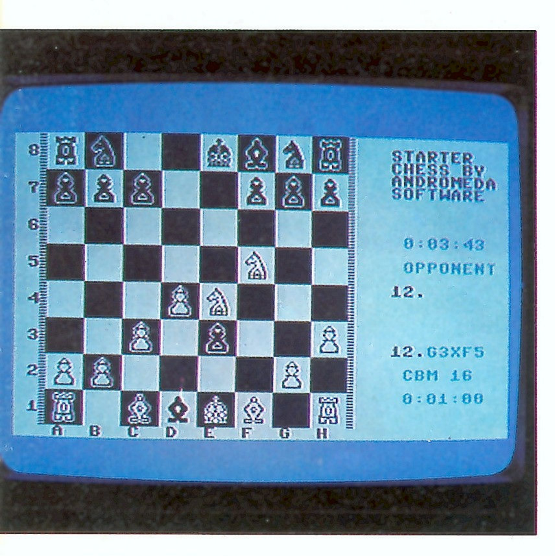
Colossus utiliza dos pantallas distintas para ofrecer información al incauto usuario que haya aceptado el desafío de una partida contra este genial programa. La primera pantalla es una representación gráfica del tablero (en dos o tres dimensiones según se haya seleccionado) y varios mensajes, además de una serie de números y letras dispuestas en los bordes del tablero (esto señala el «nombre» de cada cuadro. A2, H7, etc., como ocurre en el conocido juego de los barquitos).

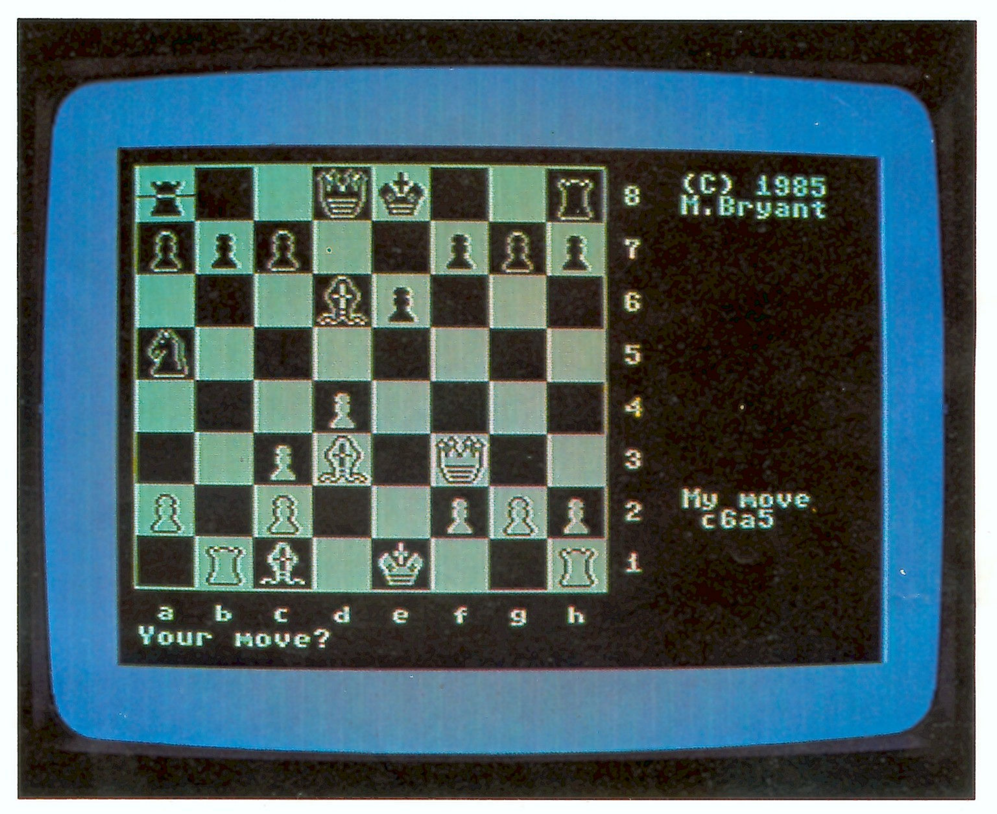
La segunda pantalla facilita los datos de la memoria del ordenador, es decir, los posibles movimientos a realizar, y también muestra los nombres de los jugadores, los relojes y los siete últimos movimientos realizados por cada bando. Dentro de los últimos movimientos a realizar por el ordenador encontramos previsiones, posiciones examinadas, línea actual y mejor línea. Esto es así, debido a que el programa utiliza nuestro tiempo para pensar, aplicando las técnicas de interrupción de







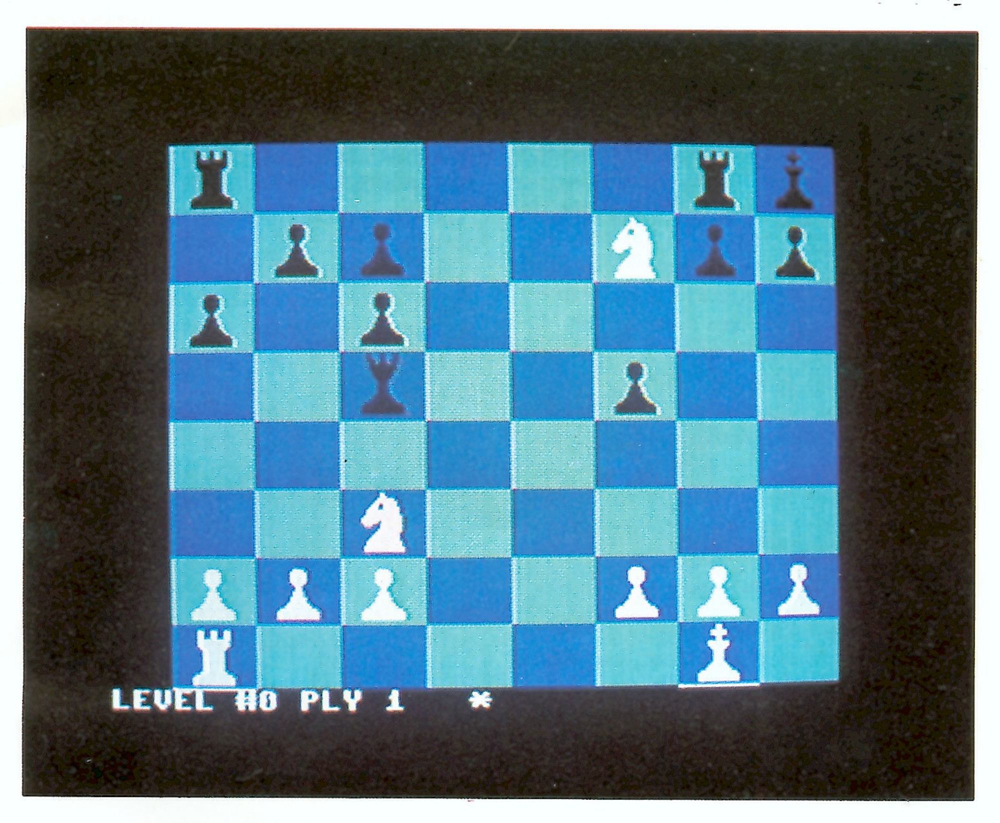


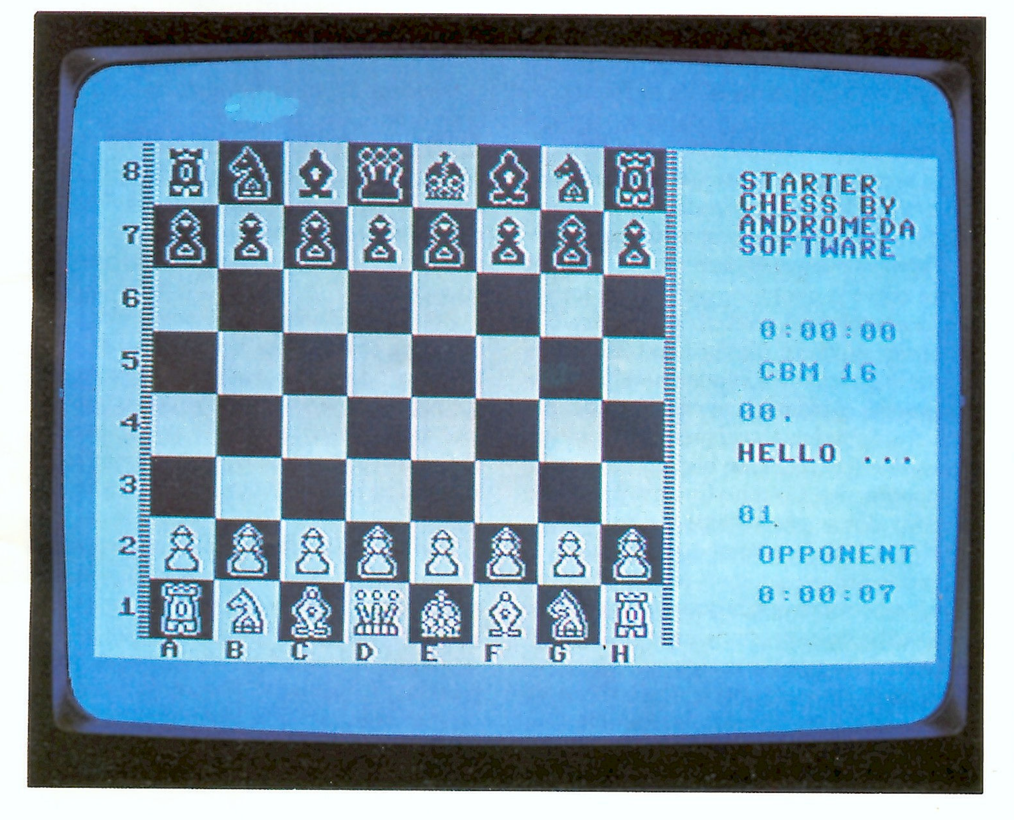


los ordenadores Commodore. Para cambiar de pantalla pulsar la tecla de espacio.

Sobre la pantalla gráfica del tablero, el ordenador muestra dos tipos de mensajes: cuando tú mueves, o cuando él mueve. El primero de ellos (Your move?), aparece en el momento en el cual debes realizar tu jugada. La forma de introducirla mediante el teclado consiste en utilizar las teclas de cursores para situar la línea horizontal (cursor) sobre la pieza a mover, pulsar **RETURN**, volver a colocar la línea sobre el cuadro de destino (donde queremos colocar esa pieza), y pulsar otra vez **RETURN.** Si un peón llega a la octava línea es posible cambiarlo por cualquier pieza (excepto un rey), en cuyo caso el programa pregunta (Promotion to?), y debemos indicar la letra correspondiente a la pieza que nos interese (N=caballo, B=alfil, R=torre y Q=reina).

Por otra parte, cuando el ordenador juegue mostrará bajo el tablero un mensaje (Let me think...), y así podremos consultar sus ideas en la pantalla de datos, a través de los diversos registros de la misma. Al acabar el proceso ejecutará lo pensado sobre el tablero, dándonos la oportunidad de encontrar una réplica convincente que nos conduzca hacia la victoria (siempre que el Colossus no haya previsto un mate en X jugadas). Si deseamos jugar cuanto antes, interrumpiremos





COLOSSUS CHESS 4-0

Nombre:	Colossus Chess 4-0
Precio:	2.912 ptas.
Soporte:	Cinta
Tipo:	Ajedrez
Ordenador:	Commodore 64 ó 128

VEREDICTO FINAL

Gráficos:	***
Sonido:	*
Puntuación:	*

SARGON II CHESS

Nombre:	Sargon II Chess
Precio:	2.500 ptas.
Soporte:	Cinta o disco
Tipo:	Ajedrez
Ordenador:	Commodore 64 ó 128

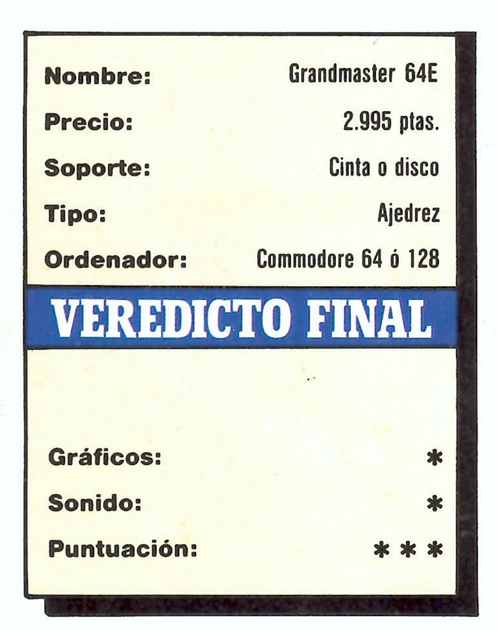
VEREDICTO FINAL

Gráficos:	**
Sonido:	**
Puntuación:	****

el proceso de pensamiento pulsando la tecla Commodore.

El programa es extraordinariamente versátil, es decir, permite ajustar el juego a nuestras necesidades y gustos, a través de una serie de teclas. Por ejemplo, **SHIFT + J** permite conectar/desconectar el joystick, **SHIFT + L** muestra todos los movimientos que puede hacer una pieza, **SHIFT + N** comienza un juego nuevo, **SHIFT + O** coloca las piezas negras arriba o abajo, con **SHIFT + P** Colossus realiza una demostración contra sí mismo

GRANDMASTER 64E

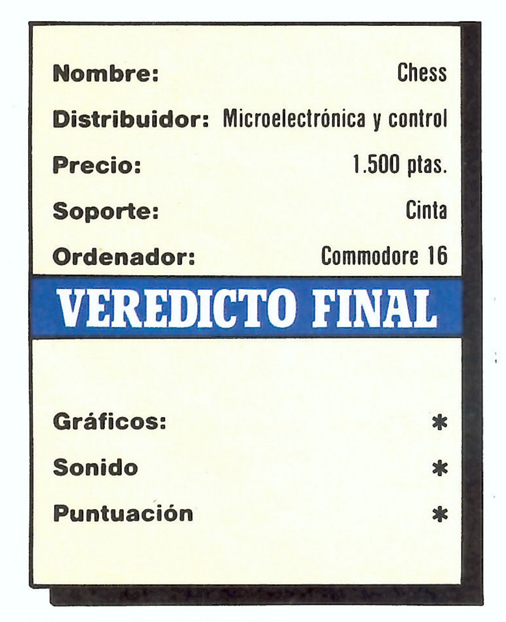


Nombre:	Grandmaster 64E
Precio:	2.995 ptas.
Soporte:	Cinta o disco
Tipo:	Ajedrez
Ordenador:	Commodore 64 ó 128

VEREDICTO FINAL

Gráficos:	*
Sonido:	*
Puntuación:	* * *

CHESS



Nombre:	Chess
Distribuidor:	Microelectrónica y control
Precio:	1.500 ptas.
Soporte:	Cinta
Ordenador:	Commodore 16

VEREDICTO FINAL

Gráficos:	*
Sonido	*
Puntuación	*

de sus propias habilidades, **SHIFT + R** *repite la última partida jugada haciendo una pausa* ajustable entre movimiento y movimiento, **SHIFT + S** para jugar los movimientos de ambas partes (para dos jugadores humanos, o para colocar la posición de los problemas).

Además de todos estos, otros como **SHIFT + T** ajusta el modo de juego, **SHIFT + U** para que el ordenador haga un movimiento distinto, **SHIFT + C** para cambiar los colores de las piezas y el tablero, **SHIFT + V** ajusta el volumen. Una última opción, la más importante de todas, permite alterar los datos y las condiciones del juego: si el ordenador utiliza su libro de posiciones, cómo juega, las dimensiones del tablero (2 ó 3), etc., etc.

Realmente, Colossus 4-0 Chess es uno de los mejores programas de ajedrez preparados para ordenadores personales. Su libro de posiciones, la capacidad para resolver problemas y ayudomates superan con mucho a cualquier otro. Si nos interesa progresar en el conocimiento del ajedrez uno de los primeros pasos a nuestro alcance es adquirir este inigualable programa.

Sargon I, II y III. El clan de los Sargon ataca

Con Sargon I los técnicos de software no consiguieron un gran programa, y se vieron obligados a crear el Sargon II, cuyas habilidades para el juego superaban ampliamente las de la anterior versión. No obstante, todo está sujeto a cambios, y el Sargon II ya no ofrece a los usuarios una gama de posibilidades adecuada, por lo cual se ha trabajado sobre los ordenadores Mackintosh consiguiendo una verdadera maravilla: el Sargon III. Desgraciadamente los ordenadores Commodore sólo pueden utilizar el Sargon II para sus periplos ajedrecísticos.

Las instrucciones de uso del programa están distribuidas en castellano, y ofrecen una información completa acerca de las características del mismo. Abarca campos tan variados como la anotación de las partidas sobre la pantalla (sin volcado en impresora o grabación en disco), o como la posibilidad de alterar la posición voluntariamente. Esto último permite el plantear un problema de ajedrez al ordenador, el cual probablemente no lo resuelva (al menos tan bien como su competidor anterior, Colossus Chess 4-0, que tarda unos dos o tres minutos en resolver un problema, por muy difícil que este sea.

Sargon II es, en resumen, una opción para aquellas personas que busquen un buen rendimiento a bajo precio. Desarrolla bien el conjunto de la partida, aunque su capacidad tanto para resolver problemas como para dar con la jugada correcta es bastante deficiente. Incorpora algunos sonidos aceptables que contribuyen decisivamente a ahuyentar la soledad que sentimos al jugar contra un ordenador (sin enfados del contrincante).

Sesenta y cuatro casillas, treinta y dos piezas, Grandmaster 64E

El manejo de este programa es sencillo, con muy pocas funciones a nuestro alcance. Las jugadas se introducen tecleando la casilla de origen (donde está la pieza), y la casilla de destino (donde queremos desplazarla). Como funciones de selección distinguimos el nivel (tecla **L**) de 0 a 9, comenzar un nuevo juego (teclas **CTRL + 0**), consultar qué jugada haría el ordenador en nuestro caso (**?**), rectificar una jugada (tecla superior izquierda), cambiar el color del tablero, de las piezas y del borde de pantalla (**F3, F5, F7** respectivamente). Asimismo, el ordenador jugará partidas contra sí mismo al pulsar **F7.**

Se trata de un programa cuya utilización no requiere muchos conocimientos, y cuya función es la de proporcionar un pasatiempo sencillo o un motivo de orgullo para el usuario. Orgullo porque hasta el nivel cuatro no es excesivamente difícil obtener una victoria, si bien los demás niveles consumen demasiado tiempo. En caso de que no nos interese perfeccionar nuestros conocimientos sobre el ajedrez, la opción en disco del Grandmaster 64E nos proporciona un simple juego, un entretenimiento nada más.

Para los usuarios del Commodore 16 el programa Chess representa una oportunidad única

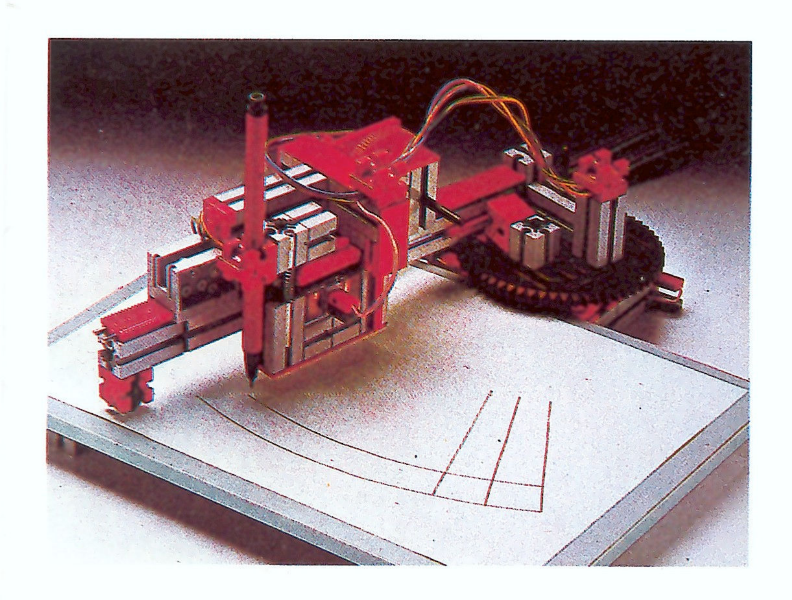
Preparado para el Commodore 16, el programa Chess tiene un manejo confuso, pues cualquier equivocación suele anular la partida en curso. Así, si en vez de pulsar la «B», pulsamos la «N» (a su lado), comenzaremos de nuevo la partida, y como las entradas deben realizarse mediante el teclado es fácil equivocarse.

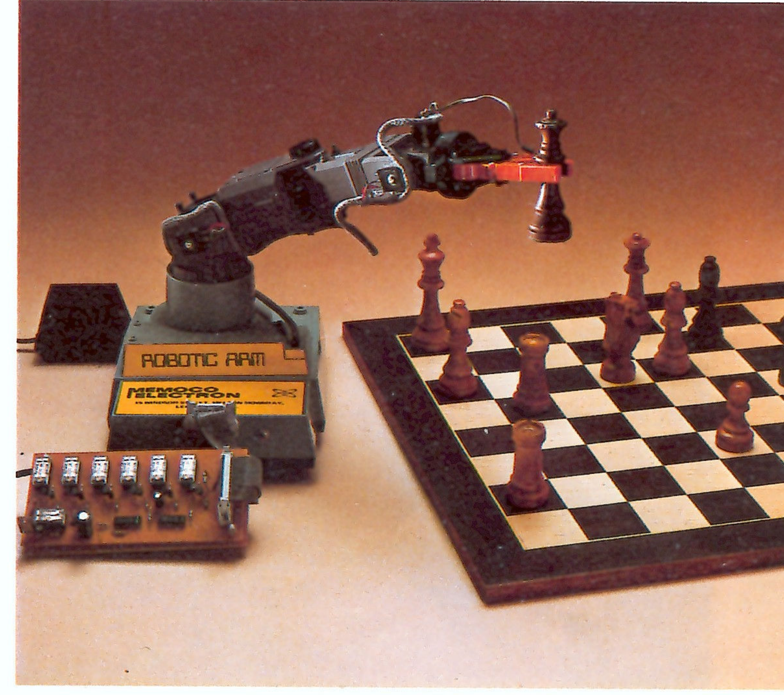
Un reducido conjunto de comandos ofrece varias posibilidades interesantes: **N** para empezar una nueva partida, **R** sitúa las piezas negras arriba o abajo, **P** ofrece una demostración del «buen» hacer del ordenador, **ESC** retrocede un movimiento, **S** memoriza la posición para volver a ella más tarde al pulsar **K.**

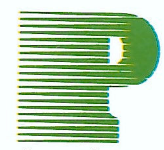
Con 16 Kbytes de memoria no son muchas las cosas que pueden hacerse. Aún así, el programa CHESS presenta muchos inconvenientes tales como la carencia de niveles de juego o la lentitud con que transcurren las partidas. De momento, es el único programa del mercado para este ordenador, quizás alguna casa se decida a crear otro en un futuro no muy lejano.

Robótica para Commodore

Hay infinidad de periféricos que se pueden conectar al C64, pero seguramente el más infrecuente es un robot. Ahora bien, infrecuente no quiere decir imposible.







Pero antes de nada, ¿qué es un robot? Bien, lo podríamos definir como un dispositivo capaz de ejecutar ciertas tareas, sobre todo motrices, de una forma lógica, inteligente por medio de programación o susceptible de programarse, que tiende a emular el movimiento humano.

No obstante, esta imitación del movimiento humano, es sumamente compleja; de ahí que los robots hayan evolucionado mucho en este sentido. Todos hemos visto alguna vez los robots de soldadura o pintado de una fábrica de automóviles; en estos casos, la semejanza humana se limita a una extremidad: el brazo.

Hay que decir, que esto tiene un sentido totalmente práctico, ya que ¿para qué necesitamos una cabeza o unas piernas en este tipo de mecanismos? También es cierto que todos hemos visto los clásicos androides de películas o juguete, en donde no falta de nada: cabeza, ojos, boca, piernas, brazos, etc., pero hoy por hoy, no dejan de ser más que eso: una ficción o un juguete muy bonito y nada más.

MEMOCO y FISCHERTECHNIK: dos robots para nuestro Commodore

En el presente artículo vamos a tratar dos robots en concreto implementables en nuestro Commodore. Por un lado el brazo MEMOCO ELECTRON, y por otro el robot desmontable FISCHERTECHNIK.

En primer lugar, hablaremos del MEMOCO ELECTRON, que pertenece al primer grupo de robots, en concreto es un brazo electro-mecánico denominado: MEMOCO ELECTRON ROBOTIC ARM, y distribuido en España por MACROCHIP, S.A. En esta clase de periféricos, es el único disponible en el mercado nacional.

A primera vista podremos preguntarnos: ¿muy bonito pero, para qué me puede servir? Está claro que a esta cuestión, debe responderse cada uno por sí mismo, pero tengamos en cuenta que la robótica es una disciplina que se encuentra en pleno desarrollo y que, queramos o no, acabará por imponerse en nuestras vidas, de igual manera que lo han hecho los ordenadores. Los robots pueden efectuar por nosotros tareas peligrosas como tratamiento de productos tóxicos o desactivación de explosivos, repetitivas como el pintado de un automóvil y tantas otras de nuestra vida ordinaria o simplemente aburridas como puede ser lavar los platos.

Con el robots MEMOCO, podremos iniciarnos en el apasionante mundo de la robótica, aprendiendo a manejar un brazo artificial, de la misma manera que manejamos el nuestro. A primera vista parece fácil, pero una vez metidos en faena, observamos que no lo es tanto; nosotros manejamos los brazos sin un esfuerzo consciente de ello mientras que ahora, debemos hacer un esfuerzo por realizar lo obvio.



Control del MEMOCO ELECTRON por programa

El brazo MEMOCO, dispone de todas las articulaciones necesarias para simular un brazo humano, estas son:

HOMBRO: Consiste en la base del brazo, esta base puede girar a izquierda y derecha, así como levantar o bajar todo el conjunto igual que el hombro.

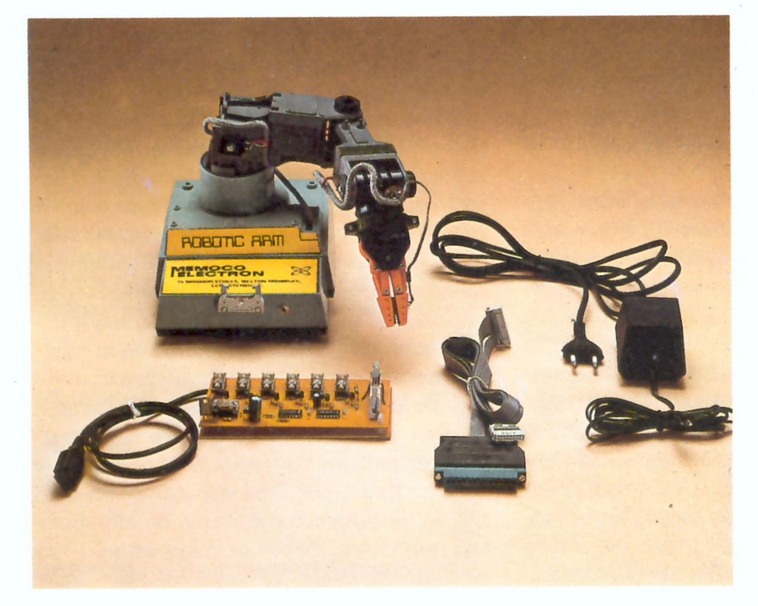
CODO: Esta es la articulación siguiente, al igual que su homóloga verdadera, dispone de un giro de izquierda y derecha.

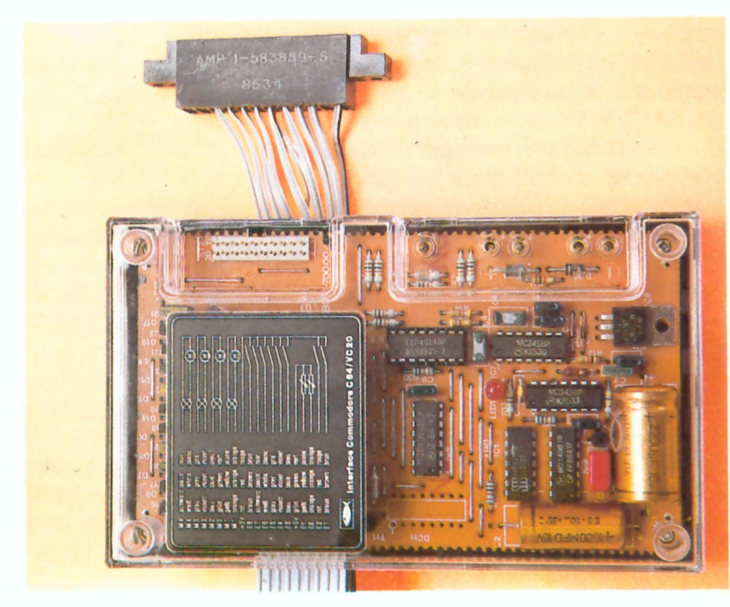
MUÑECA: Aquí olvidamos un movimiento de la muñeca, el de izquierda derecha, pero disponemos de otros dos que dan una libertad excelente: arriba y abajo, y rotación izquierda y derecha.

PINZA: En este caso, la versión mecánica sale claramente perdiendo, respecto a la humana, en lugar de la enorme versatilidad de nuestros cinco dedos, nos tendremos que conformar con dos de ellos rígidos en forma de pinza, que sólo podremos abrir o cerrar.

La forma de conexión brazo-ordenador, es un poquito complicada: En primer lugar habrá que conectar una clavija paralelo al port del usuario, a continuación una tipo joystick en el port uno, y finalmente, un pequeño transformador a la red y al robot, darán vida a los seis motores que mueven el conjunto. En todo ello, se aprecian un par de defectos, el primero es que el cableado del port del usuario es un poco corto desde la base hasta una placa intermedia, que siempre ha de quedar colgando y la segunda, es que dicha placa, no lleva ninguna protección por lo que los «reles» que lleva así como el resto de sus componentes, quedan al descubierto.

Todos los controles se efectúan por medio de POKE, pudiendo determinar la posición del brazo por medio de una corta rutina en lenguaje máquina que viene incluida en el programa de demostración, esta rutina se encarga de leer los valores de la resistencia y colocarlos en una posición de memoria. No obstante, la precisión de la unidad probada no es precisamente brillante, por lo que no





podemos pretender hacer trabajos de relojería con el brazo, pero sí es lo suficientemente exacta para manejar objetos del tamaño de una ficha de ajedrez y de dejarla en su correspondiente casilla (más o menos). De todas formas, un buen montaje con motores de precisión y toda la electro mecánica necesaria, costaría bastante más que el C64 + unidad de disco + monitor.

El manual es bastante pobre, limitándose su utilidad a la tabla de pokes necesaria para su uso. Con el brazo se acompaña una cinta de demostración que no es tampoco ninguna maravilla. No hemos encontrado el programa para mover el brazo con un joystick, así pues nos hemos confeccionado uno rápido, para manejar con el teclado, que encontrarás en las páginas de este artículo.

```
1 PRINT"{CLR} {RON}K{POF} GIRO IZQUIERDA[4 ESP]{RON}:
{ROF} GIRO DERECHA"                                    -158-
2 PRINT" {RON}O{POF} BRAZO ARRIBA[6 ESP]{RON}.{ROF} B
RAZO ABAJO"                                            -082-
3 PRINT" {RON}A{POF} MANO IZQUIERDA[4 ESP]{RON}D{ROF}
 MANO DERECHA"                                         -001-
4 PRINT" {RON}W{ROF} MANO ARRIBA[7 ESP]{RON}X{ROF} MA
NO ABAJO"                                              -000-
5 PRINT" {RON}G{ROF} CODO IZQUIERDA[4 ESP]{RON}H{ROF}
 CODO DERECHA"                                         -001-
6 PRINT"{RON}F7{ROF} CIERRA PINZA[5 ESP]{RON}F1{ROF}
ABRE PINZA":C=49152                                    -022-
10 GETX$:IFX$=""THEN10                                 -051-
15 RESTORE                                             -170-
20 FORI=0TO14                                          -039-
30 READA$,A                                            -113-
40 IFX$=A$THEN70                                       -199-
50 NEXTI                                               -013-
60 GOTO10                                              -032-
70 POKE56577,A                                         -049-
80 POKE56579,X                                         -075-
85 X=(-(X=0))*255                                      -124-
90 IFPEEK(197)<>64THEN80                               -170-
100 POKE56579,0                                        -076-
110 GOTO10                                             -076-
200 DATA _,0                                           -167-
210 DATA 1,1                                           -123-
220 DATA 2,2                                           -126-
230 DATA A,4                                           -144-
240 DATA D,5                                           -149-
250 DATA W,17                                          -220-
260 DATA X,16                                          -221-
270 DATA G,128                                         -001-
280 DATA H,129                                         -004-
290 DATA K,8                                           -164-
300 DATA ":",9                                         -208-
310 DATA O,33                                          -207-
320 DATA ".",32                                        -242-
330 DATA "{F1 }",64                                    -079-
340 DATA "{F7 }",65                                    -084-
```

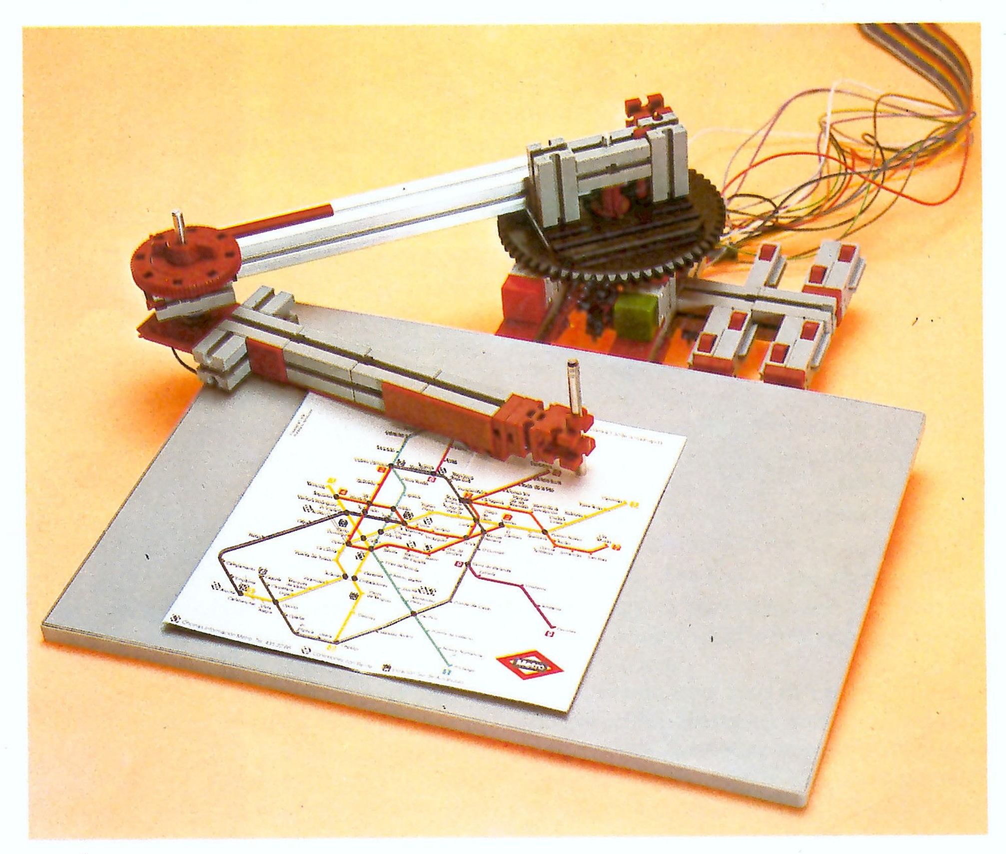
El Kit de montaje Fischertechnik

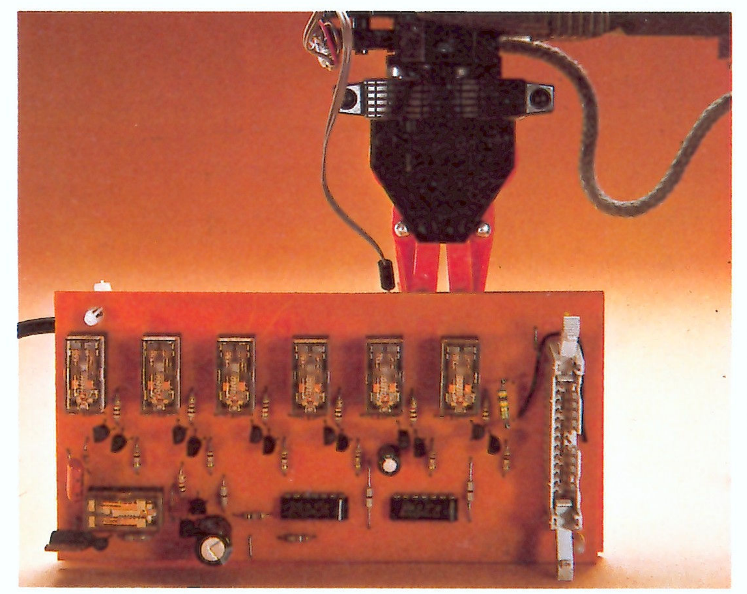
Pero, ¿qué es esto?, es lo que suele exclamar todo aquél que ve por primera vez el kit de montaje Fischertechnik, la respuesta no se deja esperar: es un mecano informatizado para varios modelos de ordenadores, entre los que se encuentra ¡cómo no!, nuestro querido Commodore.

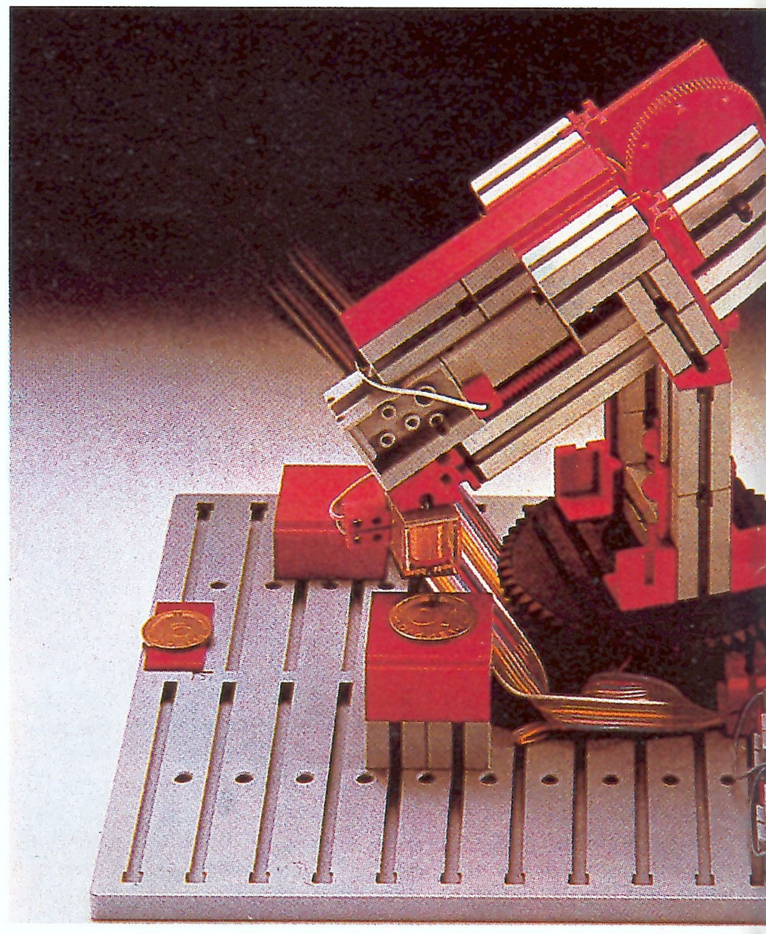
Se trata en realidad de un kit de montaje que obedece al clásico: «móntelo usted mismo! Conectado al PORT DEL USUARIO del Commodore (64 ó VIC-20) y acompañado del correspondiente programa, será el ordenador el que tome el control de todo el montaje y lo haga funcionar, pero comencemos por el principio.

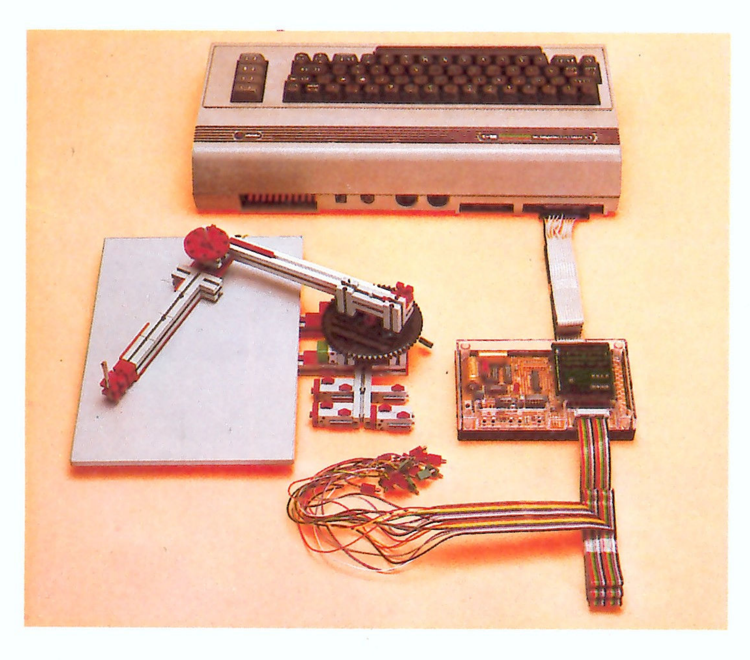
Todo el sistema viene suministrado en dos cajas con una presentación impecable, al abrirlo da verdadera pena tener que romper los plásticos, pelar los cables o desordenar aquello; pero por otro lado, la curiosidad nos corroe y nos abalanzamos sobre ello para intentar montar algo. En una de las cajas, vienen todas las piezas de montaje y en la otra el «interface» para la conexión al ordenador, pero como ya nos hemos decidido a abrir la caja de las piezas, vamos a ver lo que contiene; lo primero que llama la atención, es que el material rígido de plástico de las piezas es muy bueno, y que la facilidad de ensamblaje entre unas y otras es excepcional (eso sí, hay que acostumbrarse), un completo y bien pensado sistema de cuadradillos, acanaladuras, salientes circulares y triangulares, convierten a cada pieza en un eslabón virtual para cualquier otra; en una palabra, todas encajan con todas.

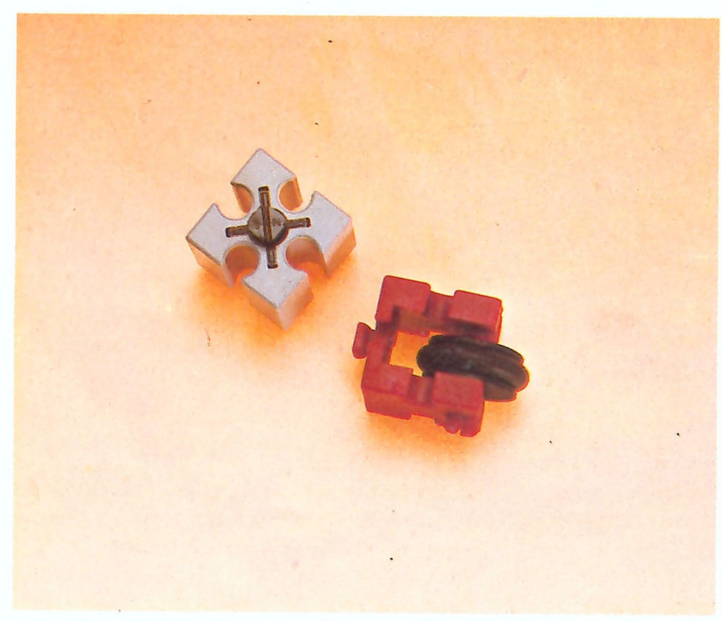
Hay todo un extenso surtido de piezas para que se pueda construir cualquier cosa imaginable, y hasta algunas cuya misión fundamental es la de adornar el conjunto final, con lo que los montajes terminados alcanzan un grado estético francamente notable (sobre todo si colocamos los cables como Dios manda).

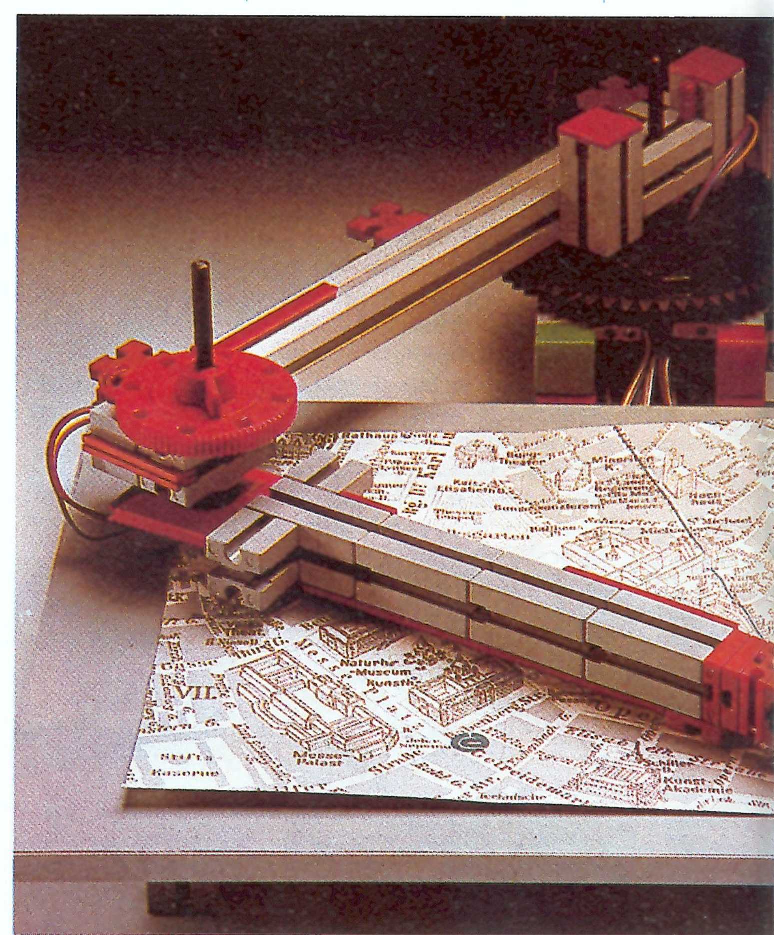


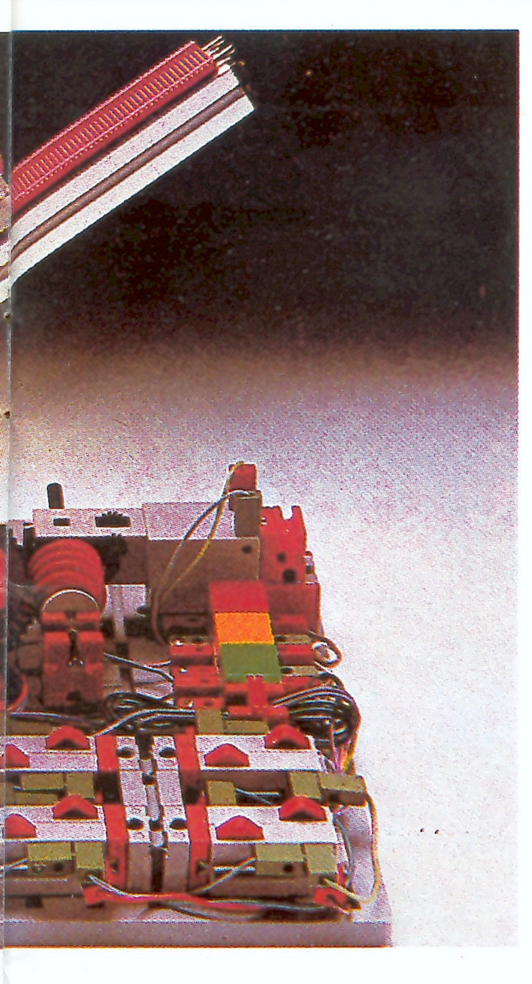


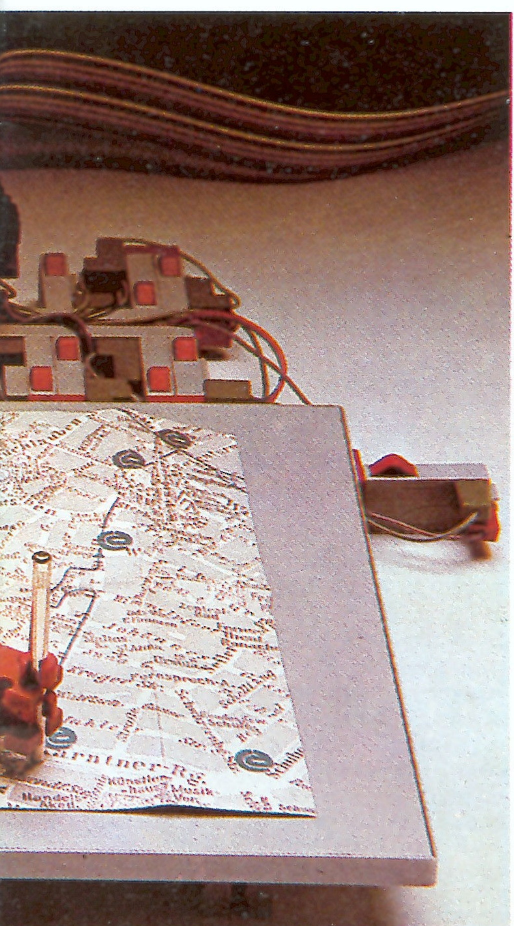












Todos los montajes se realizan sobre una placa base que contiene 20 railes equidistantes, sobre los cuales se montarán las distintas piezas de las que conste el conjunto, además, sirve para acometer el montaje correctamente colocando las primeras piezas en los lugares adecuados. Muchas de las piezas son simplemente los eslabones de la construcción, pero hay una serie de ellas, que son fundamentales para el desarrollo de los distintos diseños:

8 interruptores. Consiguen parte de la inteligencia final.

2 Motores. Dan la fuerza necesaria para generar los movimientos.

2 Potenciómetros. Esta es la segunda parte de la inteligencia final.

3 Portalámparas. Una luz siempre es bienvenida para saber lo que se está haciendo en un momento dado.

1 Electroimán. Suministra la garra necesaria.

2 Cajas de engranajes. Sirven para moderar la velocidad propia de los motores, y patras aumentar su potencia por desmultiplicación.

1 Tornillo sin fin. Para conseguir que gire...

1 Rueda dentada. Que es la base de muchos montajes.

La «interface» que se encarga del control de todo esto, viene sobre una carcasa de plástico rígido y transparente que la hace muy estética (hay otras que mejor no hablar) y tiene dos entradas y una salida; una de las entradas es la que viene directamente del PORT DEL USUARIO de Commodore, la otra, es la propia fuente de alimentación; este aspecto del equipo junto con unos manuales perfectos pero en alemán, forman el handicap que tiene que superar.

Quizás el distribuidor esté realizando ya alguna gestión para superarlo, pues junto con el disco de programas original (en alemán), se acompaña otro totalmente en castellano; en cuanto a la fuente de alimentación, el problema es que ¡no existe!, lo cual desmerece un poco la ilusión inicial que se ha puesto en el KIT, es inaudito que un equipo como éste pueda necesitar la ayuda de otras cosas para funcionar. No obstante, cualquier transformador pequeño (del tipo usado para las calculadoras), que entregue de 6 a 10 voltios de corriente continua, servirá sin ningún problema (la «interface» se encarga de ello).

Ante esto, se podría argumentar que en el port del usuario, tenemos un voltaje a nuestra disposición, es cierto, pero el tomar corriente de allí puede originar problemas en cuanto a sobrecargas, que podrían dañar irremisiblemente el ordenador (¡mucho cuidado al usarlo por nuestra cuenta!); de esta manera, se obvia un problema que podría convertir el kit en peligroso y lo hace muy agradable de utilizar, siendo la propia «interface» la encargada de que no se produzcan daños. En cuanto a la salida de la «interface» consiste en un cable paralelo de 20 hilos, que son los que llevarán la alimentación y la información necesarias para que el montaje cobre vida propia.

La utilidad del Fischertchnik

Todo esto está muy bien, pero ¿para qué sirve? ¿Qué montajes podemos realizar? En primer lugar hay que decir que son necesarios unos conocimientos básicos de electricidad (saber cablear una bombilla, montajes en paralelo y cosas así); también es necesaria una comprensión visual y espacial, ya que los montajes se realizarán siguiendo fotos paso a paso hasta llegar al resultado final.

Sobre esto hay que decir que el último diseño no viene documentado paso a paso, sino que sólo hay una foto de conjunto, y que el primero carece del correspondiente esquema de conexiones. Estos esquemas son clarísimos y sirven para cablear el conjunto adecuadamente.

El kit que hemos examinado es digamos el número uno (menos completo) y hay otros más grandes con los que los diseños se pueden ampliar (y complicar). Los diseños son ejemplos propuestos cuyos correspondientes programas están el el disco suministrado, pero sólo son ejemplos de lo que se puede hacer con un poco de práctica. Los más representativos son:

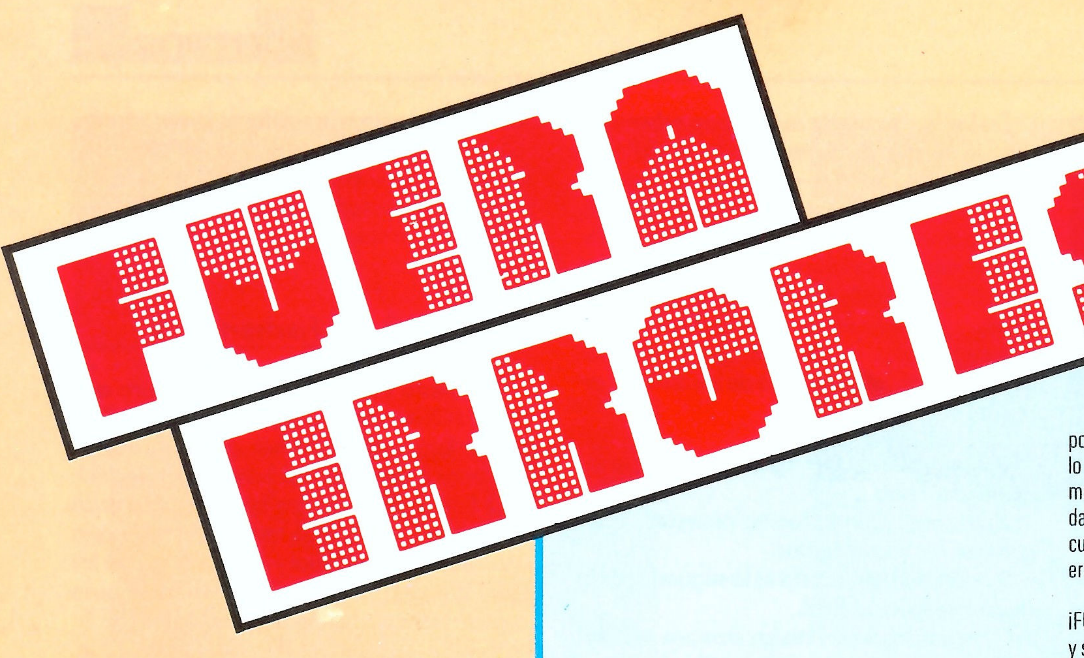
SEMAFORO: Es el más fácil de montar (10—20 minutos) y con su programa va cambiando las tres luces junto con un mensaje en la pantalla.

ROBOT: Consiste en un brazo con dos movimientos y un electroimán para coger piezas metálicas, lo bueno de este montaje es que se puede poner en modo aprendizaje para después ejecutar independientemente los movimientos realizados con anterioridad. Muy interesante.

TABLETA GRAFICA: Para este montaje y para el siguiente, es necesario disponer de la ayuda de programación SIMON'S BASIC. Consiste en un brazo articulado y un tablero; lo que dibujemos con el brazo será representado en alta resolución. Mediante interruptores (y luces) podemos elegir cuadrados, círculos, rectas, carácteres, etc.

PLOTTER: Mediante órdenes haremos dibujar sobre el tablero lo que deseemos por medio de un rotulador (rotring) suministrado con el equipo.

En definitiva se trata de un kit interesante que se podría utilizar con propósitos educativo-prácticos. Si eres un «manitas» puedes desatar su imaginación y realizar cosas sorprendentes, si no has quedado suficientemente sorprendido por el equipo en sí...



C-Byte tiene el honor de presentar, en exclusiva nacional para los lectores de TU MICRO COMMODORE, el revolucionario sistema de introducción de programas FUERA ERRORES. Este nos permitirá introducir, sin temor alguno al esfuerzo inútil, cualquier listado por largo y complicado que parezca.

Para adoptar los listados publicados bajo este sistema, deberemos seguir las siguientes normas:

1) Es fundamental transcribir EXACTAMENTE el listado reproducido, incluyendo todos sus espacios, aunque se trate de separaciones entre número de instrucción y línea de instrucción.

2) Todas las líneas finalizarán con un número de tres dígitos, encerrado entre guiones, que NO deberá ser introducido, puesto que no forma parte del programa, sino que tiene la finalidad de hacer funcionar el sistema FUERA ERRORES, según veremos más adelante. Para evitar equivocaciones, dicha cifra entre guiones se sitúa en el margen derecho del final de la línea BASIC a la cual corresponde, a una distancia prudencial del mismo.

3) Para facilitar la introducción de símbolos difícilmente interpretables, se procede a la siguiente representación en los listados.

– Las letras aparecidas entre menor y mayor deberán ser introducidas con pulsación simultánea de la tecla COMMODORE y la letra representada. Ej.: < M > =COMMODORE M.

– Las letras aparecidas entre barras verticales deberán ser introducidas como pulsación simultánea de la tecla SHIFT y la letra representada. Ej.:|K|= SHIFT K.

– Entre corchetes simples se representarán los símbolos que se obtienen por pulsación directa de la tecla, aunque lógicamente, este caso sólo se dará para indicar las sucesiones de más de una letra. Así por ejemplo, la introducción de 5 asteriscos se representaría por [5*].

– Para la repetición de símbolos obtenidos mediante las teclas COMMODORE o SHIFT, se seguirá una combinación de las tres normas anteriormente citadas. Así por ejemplo, la introducción de 10 símbolos COMMODORE H, se representaría por [< 10 H >].

– Para evitar confusiones, cuando se utilice el sistema de representación de sucesiones de carácter, y éste sea un espacio, se utilizará la abreviatura ESP. [15 ESP] = 15 espacios.

– Los caracteres de control, tales como desplazamientos del cursor, colores, estados de reversa y funciones, se simbolizarán por una abreviatura de tres letras (dos más un espacio en el caso de las funciones) encerradas entre llaves, tal como se señala en la tabla adjunta.

Para introducir cualquier listado por el sistema FUERA ERRORES, deberemos entrar previamente y ejecutar el listado BASIC que aparece en la página siguiente por lo cual es recomendable conservar una copia grabada del mismo, para sucesivas ocasiones. Una vez introducido este listado, ya sea por el teclado, o a través de cinta o disco, debemos ejecutarlo con **RUN**. Instantes más tarde aparecerá en la pantalla el mensaje **FUERA ERRORES!** y el cursor libre para la entrada de programas, con el tradicional **READY**. Por encima, lo cual indicará la activación del sistema de depuración de errores.

En virtud al **NEW** que finaliza línea 20 del programa ¡FUERA ERRORES!, éste habrá desaparecido de la memoria, y seremos libres para introducir cualquiera de los programas listados en la sección **TECLA A TECLA** de cualquier número de nuestra revista, o incluso aquellos que apareciendo en otras secciones se acojan a este sistema. Así pues, si el programa ¡FUERA ERRORES! ha desaparecido de la memoria, ¿qué hemos conseguido ejecutándolo? Bien, la respuesta se llama informáticamente **INTERRUPCION**: se trata de una técnica de programación en código máquina que permite que el ordenador efectúe prácticamente dos trabajos a un tiempo, o más correctamente, que ejecute determinada tarea de forma automática, sin necesidad de que le prestemos una atención constante, de forma similar al proceso de respiración en un humano.

Efectivamente, aunque el soporte BASIC ha desaparecido de la memoria, antes de ‹marcharse› ha dejado funcionando en modo interrupción la pequeña rutina en código máquina que se hallaba en sus **DATAS**. Para comprobarlo pulsa **RETURN**; observarás algo muy extraño: tu ordenador no se comporta normalmente, no sólo desciende una línea el cursor, sino que además hace aparecer un número en la esquina superior izquierda de la pantalla. A continuación veremos como emplearlo.

Cada vez que pulsemos **RETURN**, aparecerá un número en la mencionada zona de la pantalla, y éste corresponderá con la instrucción que hayamos introducido. Esto forma parte del sistema de FUERA ERRORES. Cuando introduzcamos cualquier línea de un listado de este tipo, deberemos fijarnos en el número que aparece al pulsar **RETURN** de fin de línea; si éste coincide con el que aparece en el listado al final de la línea, ésta habrá sido introducida correctamente, en caso contrario existe algún error de tecleado que debemos modificar. Para modificar una instrucción errónea, no tenemos ni tan siquiera que volver a teclearla si no queremos, bastará sencillamente con modificar el carácter o caracteres erróneos como siempre hacemos, hasta que coincida el número de verificación que se presentará al pulsar **RETURN**.

Así pues, el sistema FUERA ERRORES se compone de dos partes: una codificación especial de los listados que facilitan su introducción, evitando los errores al confundir los caracteres gráficos, de control, etc., y un sistema de verificación de líneas que nos advierte en el preciso instante de introducir una de estas, que está mal tecleada.

Ahora bien, el empleo de estos dos sistemas no quiere decir que nos encontremos ante un BASIC diferente al de COMMODORE 64. Este no ha cambiado, hace exactamente las mismas cosas de siempre; simplemente hemos cambiado la forma de hacer los listados. En cuanto al misterioso número que aparece en la esquina de la pantalla no es más que una simple suma de comprobación, lo que se conoce técnicamente como un CHECKSUM. La rutina en código máquina de interrupción suma los valores de los caracteres que entra-

TABLA DE INTERPRETACION DE CODIGOS DE CONTROL

ABR	SIFNIFICADO	OBTENCION	ABR	SIFNIFICADO	OBTENCION
HOM	HOME	CLR/HOME	F8	FUNCION 8	SHIFT F7
CLR	CLEAR + HOME	SHIFT CLR/HOME	BLK	BLACK (NEGRO)	CTRL 1
ABJ	CURSOR ABAJO	CRSR VERTICAL	WHT	WHITE (BLANCO)	CTRL 2
ARB	CURSOR ARRIBA	SHIFT CRSR VERTICAL	RED	RED (ROJO)	CTRL 3
DCH	CURSOR DERECHA	CRSR HORIZONTAL	CYN	CYAN (CIAN)	CTRL 4
IZQ	CURSOR IZQUIERDA	SHIFT CSRS HORIZONTAL	PUR	PURPPLE (PURPURA)	CTRL 5
RON	REVERSE ON	CTRL 9	GRN	GREEN (VERDE)	CTRL 6
ROF	REVERSE OFF	CTRL 0	BLU	BLUE (AZUL)	CTRL 7
F1	FUNCION 1	F1	YEL	YELLOW (AMARILLO)	CTRL 8
F2	FUNCION 2	SHIFT F1	NRJ	NARANJA	COMMODORE 1
F3	FUNCION 3	F3	MRN	MARRON	COMMODORE 2
F4	FUNCION 4	SHIFT F3	RCL	ROJO CLARO	COMMODORE 3
F5	FUNCION 5	F5	GR1	GRIS 1	COMMODORE 4
F6	FUNCION 6	SHIFT F5	GR2	GRIS 2	COMMODORE 5
F7	FUNCION 7	F7	VCL	VERDE CLARO	COMMODORE 6
			ACL	AZUL CLARO	COMMODORE 7
			GR3	GRIS 3	COMMODORE 8

mos en cada línea que introducimos, aplicándoles un módulo 256, es decir, volviendo a cero cada vez que su suma parcial supera el 255, de este modo se crea un número entre 0 y 255 dependiente directamente de los caracteres introducidos. Así pues, es prácticamente imposible que una sucesión de errores den por casualidad ese número, mientras que siempre que la línea esté correctamente introducida obtendremos el mismo código que nosotros hallamos por igual sistema y añadimos al final de cada línea al realizar el listado del programa.

Por tanto, el misterioso número no es tampoco una modificación del BASIC de COMMODORE, sino simplemente un pequeño truco para la comprobación de que las líneas han sido bien introducidas. Propiamente no nos evita cometer errores de tecleado, sino que simplemente nos advierte inmediatamente en qué línea los hemos introducido.

Para desactivar el sistema sólo deberemos pulsar **RUN/STOP+RESTORE**, y si por cualquier motivo nos interesara reactivarlo, podríamos ejecutar **SYS 822**, siempre y cuando se encuentre el código máquina en la memoria, lógicamente.

¡ADVERTENCIA! Puesto que el código máquina se encuentra ubicado en el buffer del casete, es imprescindible desactivarlo **(RUN/STOP+RESTORE)** antes de realizar cualquier operación con dicho periférico.

Utilización del casete con fuera errores

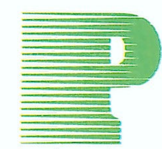
Si queremos introducir parte de un programa, para continuar posteriormente el trabajo emprendido, sin perder por supuesto la enorme ventaja del FUERA ERRORES deberemos proceder de la siguiente forma:

1. Desactivar el sistema FUERA ERRORES mediante la pulsación de RUN/STOP + RESTORE.
2. Trasladar el código máquina desde su ubicación en el buffer del casete hasta otro punto desocupado en la memoria RAM (por ejemplo 49152). Para ello sólo hemos de saber que ocupa 114 bytes desde 822 (inclusive). Una sencilla rutina que efectúe el trabajo de reubicación puede ser: **FORI=0TO113:POKE49152+I,PEEK(822+I):NEXT.**
3. Realizar la operación correspondiente con el casete; ya sea grabar una copia de seguridad de lo introducido en la memoria hasta el momento, ya sea cargar el casete una parte ya grabada del programa.
4. Restablecer FUERA ERRORES a su punto original. Siguiendo el caso de la línea de ejemplo anterior: **FORI=0TO113:POKE822+I,PEEK(49152+I):NEXT.**
5. Reactivar el sistema FUERA ERRORES mediante SYS 822 (esta vez no aparecerá mensaje alguno, aunque al pulsar **RETURN** comprobaremos que aparecen las cifras de control en la esquina superior izquierda). Lógicamente, los pasos 4 y 5 no son necesarios si después de una grabación se va a apagar el ordenador, y sólo se llevarán a cabo después de las cargas, o si tras la grabación de una copia de seguridad del programa, deseamos continuar introduciéndolo acto seguido.

```
10 FORI=822TO935:READA:C=C+A:POKEI,A:NEXT
20 IFC<>15254THENPRINTCHR$(147)"ATENCION!, HAY UN ERROR EN LOS DATOS":END
30 PRINTCHR$(147)TAB(213)"FUERA ERRORES!":SYS822:NEW
100 DATA 169,3,141,37,3,169,69,141,36,3,169,0,133,254,96,32,87,241,133,251
110 DATA 134,252,132,253,8,201,13,240,13,24,101,254,133,254,165,251,166,252
120 DATA 164,253,40,96,169,13,32,210,255,165,214,141,176,3,206,176,3,169,0
130 DATA 133,216,169,18,32,210,255,169,19,32,210,255,169,45,32,210,255,166
140 DATA 254,224,100,176,5,169,48,32,210,255,224,10,176,5,169,48,32,210,255
150 DATA 169,0,133,254,32,205,189,169,45,32,210,255,173,176,3,133,214,76,88,3
```

Jet-Man

Verse abandonado en un asteroide perdido no es nada agradable, especialmente si ese asteroide está habitado por una raza de alienígenas Astronatófagos (con perdón). Por eso, no es de extrañar que intentemos lo imposible: llamar a nuestra mamá...



Para pedir auxilio en una situación como esta, nada mejor que una radio. Pero, atendiendo a la famosa e infalible ley de Murphy, todo lo que podía salir mal ha salido mal. Para empezar, las baterías que alimentan al transmisor se han desperdigado por la superficie del asteroide. Y para seguir, un laser que pasa nos recuerda que no estamos solos...

Deberás recorrer la superficie del asteroide, buscando las baterías. Estas se hallan representadas por unas cajitas de color negro, que para mayor mofa y escarnio tienen la facultad de confundirse entre las piedras de la superficie. De tu habilidad y buen ojo dependerá el verlas.

Estas baterías deberán ser transportadas al bloque transmisor, que como los más saga-

ces habréis supuesto, tiene forma de bloque transmisor. Depositaremos las baterías sobre el susodicho, y nos alejaremos un poco para que el sistema anticalambres se decida a conectarlas.

Cuando hallamos colocado las cinco baterías que son necesarias, el transmisor emitirá la señal de la esperanza (si, como lo del teléfono pero en plan espacial).

Los mandos son muy sencillos: Joystick para todo

Para controlar a nuestro astronauta, colocaremos un joystick en el port 2. Moviéndolo arriba y abajo variaremos nuestra altura sobre el suelo, y hacia los lados nuestra velocidad.





El botón de disparo tiene una doble función. Por un lado, controla nuestro armamento energético defensivo antialienígena (vulgo trabuco), pero además nos permite coger o dejar las baterías.

Esto último sólo podremos hacerlo cuando estemos parados sobre el suelo, es decir, con nuestro astronauta mirándonos de frente. Deberemos descender a una altura suficientemente baja, y podremos tomar (o dejar) la batería.

Al comienzo del juego, podremos optar por la aparición o no de una fuerza de gravedad. De existir, ésta nos impulsará hacia abajo (original, ¿verdad?), con el consiguiente riesgo de estrellarnos contra los cráteres y otros detalles solares (es decir, del suelo). De todas formas, no nos preocupemos; también podemos estrellarnos nosotros solitos, sin ayuda de la gravedad, con un simple «tirón de joystick».

Doble listado, doble trabajo...

El listado, debido a su longitud, se presenta dividido en dos partes. Deberemos teclear y grabar el listado 1 en primer lugar (sí, ya sabemos que os lo sabiáis...).

Tal como aparece, este programa intentará cargar la segunda parte desde el disco. Para emplear la cinta, simplemente sustituir el 8 de la última línea por un 1, número de casete.

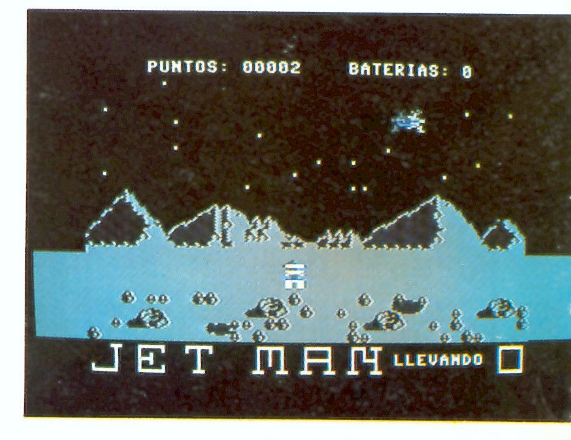
Esta primera parte se encarga de introducir en la memoria los caracteres programados, sprites y mucho código máquina, que son imprescindibles para el correcto funcionamiento del programa 2. Por tanto, el que avisa... JAMAS deberemos ejecutar el programa 2 sin haber cargado y ejecutado antes el listado 1.

El segundo listado se grabará a continuación del primero en la misma cinta o disco, con el nombre «JET 2».

Dedos hábiles, al teclado.







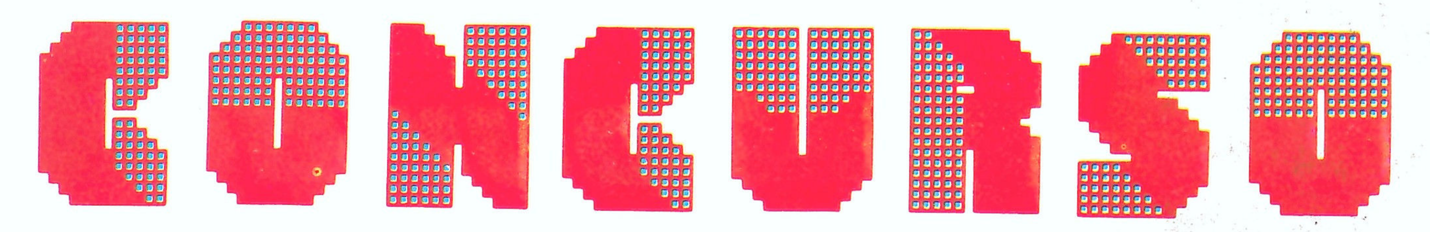
LISTADO

```
1 REM ****************************           *    JE
T MAN ; LISTADO 1      *                        -114-
2 REM *   (C) 1986 BY SINDIN SOFT   *        *    TU
 MICRO COMMODORE       *                        -142-
3 REM ****************************              -229-
4 :                                             -142-
10 PRINT"{CLR}{GRN}[10 ESP]ESPERE, POR FAVOR
                                                -192-
20 POKE53280,.:POKE53281,.:POKE56,54:CLR        -087-
30 POKE56334,PEEK(56334)AND254:POKE1,PEEK(1)AND251:J=
38912                                           -108-
40 FORI=14336TO16383:POKEI,PEEK(I+J):NEXT       -037-
50 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334)OR1
                                                -116-
60 READA:IFA=999THENPOKE53272,(PEEK(53272)AND241)OR14
:GOTO400                                        -163-
70 FORI=.TO7:READB:POKE14336+I+8*A,B:NEXT:GOTO60
                                                -000-
80 DATA141,0,0,0,0,3,4,8,8                      -047-
90 DATA142,0,0,0,60,203,5,62,195                -097-
100 DATA143,0,0,0,0,192,160,80,176              -188-
110 DATA144,25,58,188,127,254,127,174,87        -020-
120 DATA145,21,10,0,31,251,21,159,225           -082-
130 DATA146,248,116,164,194,2,12,48,192         -213-
140 DATA147,0,0,0,5,10,21,10,5                  -234-
150 DATA148,0,0,24,36,36,90,36,24               -157-
160 DATA150,200,18,105,98,136,22,70,21          -145-
170 DATA149,24,36,100,186,81,169,82,60          -168-
180 DATA201,0,60,66,66,126,110,94,60            -050-
190 DATA202,24,36,36,24,16,28,16,28             -008-
200 DATA151,255,85,170,85,170,85,170,255        -010-
210 DATA152,1,2,4,8,16,32,64,128                -101-
220 DATA153,128,128,128,128,128,128,128,128     -159-
230 DATA154,0,0,0,0,0,0,0,255                   -181-
240 DATA155,1,1,1,1,1,1,1,1                     -083-
250 DATA156,255,0,0,0,0,0,0,0                   -185-
260 DATA140,129,131,198,237,234,253,250,245     -155-
270 DATA128,128,128,200,248,252,252,254,255     -157-
280 DATA129,0,0,0,0,192,248,236,223             -252-
290 DATA130,0,192,224,252,238,245,251,255       -050-
300 DATA131,0,24,108,222,175,95,175,95          -157-
310 DATA132,175,95,175,95,175,95,175,95         -240-
320 DATA133,175,95,175,87,187,125,187,127       -079-
330 DATA134,175,95,175,87,175,87,170,85         -240-
340 DATA135,170,85,170,85,170,85,170,85         -222-
350 DATA136,85,171,213,191,255,255,255,255      -115-
360 DATA137,0,3,6,61,106,213,170,85             -000-
370 DATA138,0,0,0,0,3,15,53,234                 -042-
380 DATA139,1,1,2,5,46,125,106,213              -202-
390 DATA 999                                    -161-
400 READA:IFA=999THEN710                        -202-
410 FORI=64*ATOI+63:READA:POKEI,A:NEXT:GOTO400
                                                -173-
420 DATA221,0,0,0,0,15,0,0,23,240,0,85,252,1,85,234,1
3                                               -119-
430 DATA85,234,48,85,252,48,21,112,13,42,160,5,85,160
,5,111                                          -162-
440 DATA149,93,111,252,93,107,252,80,85,80,80,42,128,
0,10,128                                        -011-
450 DATA0,2,160,0,90,176,0,90,240,0,20,84,0,0,0,0
                                                -132-
460 DATA223,0,0,0,0,240,0,5,124,0,21,95,0,169,95,192,
169                                             -248-
470 DATA95,80,21,87,12,5,84,12,10,168,112,10,85,80,86
,249                                            -079-
480 DATA80,63,249,117,63,233,117,5,85,5,2,168,5,2,160
,0                                              -227-
490 DATA10,128,0,14,165,0,15,165,0,21,20,0,0,0,0,0
                                                -175-
500 DATA222,0,0,0,0,28,0,0,95,0,1,87,192,6,170,176,6
                                                -043-
510 DATA170,176,1,87,192,0,87,0,2,170,128,13,170,112,
14,85                                           -108-
520 DATA176,15,215,240,3,215,192,1,85,64,2,170,128,2,
130,128                                         -204-
530 DATA2,130,128,2,130,128,2,130,128,1,65,64,0,0,0,0
                                                -079-
540 DATA220,240,0,0,240,0,0,0,0,0,0,0,0,0,0,0,0
                                                -243-
550 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0         -104-
560 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0         -105-
570 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0         -106-
580 DATA219,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0     -051-
590 DATA0,0,0,0,0,0,0,0,0,20,84,0,42,170,170,85,85
                                                -090-
600 DATA84,42,170,128,85,85,0,42,170,168,17,85,80,0,1
68,0                                            -076-
610 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0         -101-
620 DATA216,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0     -043-
630 DATA66,0,1,0,128,1,0,128,1,90,128,5,86,160,95,215
                                                -112-
640 DATA250,80,212,58,5,86,160,7,255,224,17,86,136,20
,24,40                                          -170-
650 DATA16,0,8,84,0,26,0,0,0,0,0,0,0,0,0,0      -028-
660 DATA217,0,0,0,0,0,0,0,0,0,0,0,0,4,0,32,16
                                                -160-
670 DATA0,8,4,0,32,1,0,128,1,90,128,5,86,160,95,215
                                                -013-
680 DATA250,92,23,10,5,86,160,7,255,224,17,86,136,20,
24,40                                           -117-
690 DATA16,0,8,84,0,26,0,0,0,0,0,0,0,0,0,0      -032-
700 DATA218,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0     -044-
710 DATA0,0,0,0,0,0,0,0,0,170,0,21,85,64,42,170
                                                -022-
720 DATA168,85,85,84,42,170,170,85,85,84,42,170,168,5
,85,80                                          -207-
730 DATA0,170,128,0,0,0,0,0,0,0,0,0,0,0,0,0     -059-
740 DATA 999                                    -160-
750 BU=50643:FORI=BUTOBU+4*256:POKEI,32:NEXT
                                                -236-
760 DATA32,141,142,143,147,144,145,146          -159-
770 DIM C%(1,3),B%(3,9):FORI=.TO1:FORJ=.TO3:READC%(I,
J):NEXT:NEXT                                    -146-
780 DATA 32,32,32,154,154,154,154,154,154,32
                                                -181-
790 DATA 32,32,152,32,32,32,32,32,152,153       -018-
800 DATA 32,152,32,32,32,32,32,152,152,32       -009-
810 DATA 155,151,151,151,151,151,151,152,32,32
                                                -006-
820 FORI=.TO3:FORJ=.TO9:READB%(I,J):NEXT:NEXT
                                                -156-
830 FORI=1TO24:A=INT(RND(1)*2):FORJ=.TO1:FORK=.TO3
                                                -111-
840 POKEBU+256*J+256*A+10*I+K,C%(J,K):NEXT:NEXT:NEXT
                                                -119-
850 FORJ=.TO3:FORK=.TO9                         -226-
860 POKEBU+256*J+K,B%(J,K):NEXT:NEXT            -205-
870 FORI=1TO200                                 -146-
880 A=1024*RND(1):IFPEEK(BU+A)<>32THEN880       -133-
890 POKEBU+A,148+2*RND(1):NEXT                  -034-
900 FORI=49152TO50642:READA:POKEI,A:NEXT        -254-
910 DATA 120,169,0,141,14,220,169,44,141,20,3,169,192
,141,21,3                                       -021-
920 DATA 173,17,208,41,127,141,17,208,169,130,141,18,
208,169,1,141                                   -241-
930 DATA 26,208,169,0,141,64,194,141,65,194,88,96,173
,22,208,41                                      -118-
940 DATA 248,13,64,194,141,22,208,162,170,169,79,160,
192,141,20,3                                    -200-
950 DATA 140,21,3,173,25,208,141,25,208,142,18,208,76
,188,254,173                                    -200-
960 DATA 22,208,41,248,141,22,208,169,12,141,32,208,1
41,33,208,169                                   -240-
970 DATA 104,160,192,162,194,76,61,192,173,22,208,41,
248,13,65,194                                   -006-
980 DATA 141,22,208,162,226,169,124,160,192,76,61,192
,173,22,208,41                                  -046-
990 DATA 248,141,22,208,169,0,141,32,208,141,33,208,1
73,25,208,141                                   -239-
1000 DATA 25,208,169,44,141,20,3,169,192,141,21,3,169
,130,141,18                                     -126-
1010 DATA 208,173,66,194,201,0,240,6,206,66,194,76,16
8,193,173,68                                    -210-
1020 DATA 194,41,8,208,246,173,68,194,41,7,24,105,1,1
41,66,194                                       -050-
1030 DATA 173,68,194,201,128,16,113,238,64,194,173,64
,194,201,8,208                                  -048-
1040 DATA 218,169,0,141,64,194,173,183,5,141,60,3,173
,223,5,141                                      -084-
1050 DATA 61,3,173,7,6,141,62,3,173,47,6,141,63,3,173
,87                                             -255-
1060 DATA 6,141,64,3,162,38,189,144,5,157,145,5,189,1
84,5,157                                        -015-
1070 DATA 185,5,189,224,5,157,225,5,189,8,6,157,9,6,1
```

```
89,48                                                   -141-
1080 DATA 6,157,49,6,202,16,223,173,60,3,141,144,5,17
3,61,3                                                  -140-
1090 DATA 141,184,5,173,62,3,141,224,5,173,63,3,141,8
,6,173                                                  -143-
1100 DATA 64,3,141,48,6,76,168,193,206,64,194,173,64,
194,201,255                                             -160-
1110 DATA 208,243,169,7,141,64,194,173,144,5,141,60,3
,173,184,5                                              -094-
1120 DATA 141,61,3,173,224,5,141,62,3,173,8,6,141,63,
3,173                                                   -081-
1130 DATA 48,6,141,64,3,162,0,189,145,5,157,144,5,189
,185,5                                                  -162-
1140 DATA 157,184,5,189,225,5,157,224,5,189,9,6,157,8
,6,189                                                  -187-
1150 DATA 49,6,157,48,6,232,224,39,208,221,173,60,3,1
41,183,5                                                -252-
1160 DATA 173,61,3,141,223,5,173,62,3,141,7,6,173,63,
3,141                                                   -083-
1170 DATA 47,6,173,64,3,141,87,6,173,68,194,41,8,240,
3,76                                                    -069-
1180 DATA 61,194,173,67,194,201,0,240,6,206,67,194,76
,61,194,173                                             -162-
1190 DATA 68,194,41,127,141,67,194,173,68,194,201,128
,16,73,238,65                                           -019-
1200 DATA 194,173,65,194,201,8,208,215,169,0,141,65,1
94,174,69,194                                           -008-
1210 DATA 160,0,185,248,6,157,211,198,232,200,192,40,
208,244,206,69                                          -034-
1220 DATA 194,174,69,194,160,0,189,211,197,153,208,6,
189,211,198,153                                         -114-
1230 DATA 248,6,189,211,199,153,32,7,189,211,200,153,
72,7,232,200                                            -192-
1240 DATA 192,40,208,226,76,61,194,206,65,194,173,65,
194,201,255,208                                         -105-
1250 DATA 243,169,7,141,65,194,174,69,194,160,0,185,2
48,6,157,211                                            -216-
1260 DATA 198,232,200,192,40,208,244,238,69,194,76,24
1,193,76,80,194                                         -116-
1270 DATA 0,0,0,0,8,0,0,1,1,32,0,1,0,0,0,0              -248-
1280 DATA 173,70,194,201,0,240,28,206,71,194,173,71,1
94,201,0,208                                            -184-
1290 DATA 18,169,2,141,71,194,173,1,208,201,200,240,6
,238,1,208                                              -080-
1300 DATA 238,4,208,173,75,194,201,1,240,13,173,47,19
5,201,1,208                                             -131-
1310 DATA 3,76,194,195,76,251,195,173,0,220,41,15,170
,41,1,208                                               -038-
1320 DATA 16,173,1,208,201,90,240,9,206,1,208,206,4,2
08,76,179                                               -035-
1330 DATA 194,138,41,2,208,13,173,1,208,201,200,240,6
,238,1,208                                              -067-
1340 DATA 238,4,208,206,72,194,173,72,194,201,255,240
,3,76,52,195                                            -205-
1350 DATA 169,2,141,72,194,138,41,4,208,48,173,68,194
,201,128,16                                             -157-
1360 DATA 18,173,68,194,201,0,240,3,206,68,194,169,22
3,141,248,7                                             -154-
1370 DATA 76,52,195,238,68,194,173,68,194,41,8,240,71
,169,8,141                                              -133-
1380 DATA 68,194,169,222,141,248,7,76,52,195,138,41,8
,208,53,173                                             -176-
1390 DATA 68,194,41,128,240,18,173,68,194,201,128,240
,3,206,68,194                                           -008-
1400 DATA 169,221,141,248,7,76,52,195,238,68,194,173,
68,194,41,8                                             -179-
1410 DATA 240,18,169,136,141,68,194,169,222,141,248,7
,76,52,195,0                                            -211-
1420 DATA 0,0,0,0,173,47,195,201,1,208,3,76,194,195,1
73,0                                                    -032-
1430 DATA 220,41,16,240,8,169,0,141,74,194,76,251,195
,173,68,194                                             -158-
1440 DATA 41,8,240,50,173,74,194,201,1,240,239,173,1,
208,240,234                                             -128-
1450 DATA 201,180,144,230,173,10,7,201,32,240,4,201,2
00,144,219,172                                          -251-
1460 DATA 10,7,173,73,194,141,10,7,140,73,194,140,189
,7,169,1                                                -254-
1470 DATA 141,74,194,76,251,195,173,68,194,201,128,16
,9,173,0,208                                            -215-
1480 DATA 56,233,8,76,156,195,173,0,208,24,105,24,141
,2,208,173                                              -098-
1490 DATA 1,208,24,105,10,141,3,208,169,1,141,47,195,
173,68,194                                              -095-
1500 DATA 41,128,73,128,141,48,195,173,21,208,9,2,141
,21,208,76                                              -093-
1510 DATA 251,195,173,48,195,201,128,208,9,206,2,208,
206,2,208,76                                            -200-
1520 DATA 216,195,238,2,208,238,2,208,173,3,208,73,1,
141,3,208                                               -038-
1530 DATA 173,2,208,201,2,208,16,169,0,141,47,195,173
,21,208,41                                              -081-
1540 DATA 253,141,21,208,76,251,195,201,90,240,236,17
3,30,208,41,219                                         -079-
1550 DATA 141,166,2,41,2,240,16,173,21,208,41,253,141
,21,208,173                                             -119-
1560 DATA 166,2,41,253,141,167,2,173,49,195,201,1,240
,3,76,128                                               -042-
1570 DATA 196,173,50,195,201,1,240,52,173,6,208,24,10
5,3,141,6                                               -033-
1580 DATA 208,144,8,173,16,208,9,8,141,16,208,173,16,
208,41,8                                                -004-
1590 DATA 208,3,76,128,196,173,6,208,201,100,144,246,
169,0,141,49                                            -201-
1600 DATA 195,173,21,208,41,247,141,21,208,76,128,196
,173,6,208,56                                           -255-
1610 DATA 233,3,141,6,208,176,8,173,16,208,41,247,141
,16,208,173                                             -141-
1620 DATA 16,208,41,8,208,204,173,6,208,201,10,144,20
7,76,128,196                                            -190-
1630 DATA 173,49,195,201,1,208,36,173,6,208,56,237,0,
208,41,127                                              -097-
1640 DATA 201,6,176,23,173,7,208,56,237,1,208,41,127,
201,6,176                                               -041-
1650 DATA 10,234,234,169,0,141,75,194,76,171,196,173,
171,2,201,1                                             -141-
1660 DATA 240,3,76,251,196,173,168,2,141,8,208,141,10
,208,173,169                                            -201-
1670 DATA 2,201,0,240,11,173,16,208,9,48,141,16,208,7
6,216,196                                               -039-
1680 DATA 173,16,208,41,207,141,16,208,234,169,190,14
1,11,208,173,170                                        -134-
1690 DATA 2,141,9,208,238,51,195,173,51,195,201,4,208
,13,169,0                                               -048-
1700 DATA 141,51,195,173,252,7,73,1,141,252,7,173,173
,2,201,1                                                -233-
1710 DATA 240,3,76,208,197,173,171,2,201,1,176,13,173
,172,2,201                                              -078-
1720 DATA 1,240,3,76,208,197,76,164,197,173,172,2,201
,1,240,48                                               -048-
1730 DATA 169,0,162,7,157,255,211,202,208,250,169,2,1
41,78,194,169                                           -000-
1740 DATA 1,141,76,194,169,6,141,77,194,169,8,141,5,2
12,169,255                                              -120-
1750 DATA 141,6,212,169,21,141,4,212,169,1,141,172,2,
76,208,197                                              -089-
1760 DATA 173,77,194,201,6,176,30,238,77,194,173,79,1
94,141,15,212                                           -016-
1770 DATA 169,23,24,109,78,194,141,78,194,169,9,24,10
9,79,194,141                                            -239-
1780 DATA 79,194,76,208,197,238,76,194,173,76,194,201
,4,176,19,169                                           -049-
1790 DATA 50,141,79,194,169,1,141,77,194,173,78,194,1
41,1,212,76                                             -172-
1800 DATA 208,197,169,2,141,78,194,169,0,141,76,194,1
69,6,141,77                                             -178-
1810 DATA 194,76,208,197,169,0,141,172,2,141,171,2,16
2,21,157,255                                            -198-
1820 DATA 211,202,208,250,169,31,141,12,212,169,12,14
1,13,212,169,5                                          -011-
1830 DATA 141,8,212,169,0,141,7,212,169,129,141,11,21
2,76,208,197                                            -192-
1840 DATA 76,49,234                                     -242-
1850 LOAD"JET 2",8                                      -235-
1860 REM PARA CASETTE, SUSTITUIR EL 8 POR 1             -211-
```

```
1 REM ****************************                 *   JE
T MAN ; LISTADO 2      *                               -115-
2 REM *  (C) 1986 BY SINDIN SOFT  *                *   TU
MICRO COMMODORE        *                               -142-
3 REM ****************************                     -229-
4 :                                                    -142-
10 CLR:GOTO780                                         -116-
20 POKE680,50+ABS(205*SIN(3*PA))                       -028-
30 POKE682,50+ABS(100*COS(2*PA)):PA=PA+.01             -102-
40 RETURN                                              -100-
50 IFPEEK(680)<160THEN70                               -145-
60 POKEV+6,PEEK(680)-4:POKESA,1:GOTO80                 -066-
```

```
70 POKEV+6,PEEK(680)+28:POKESA,.                       -153-
80 POKEV+7,PEEK(682)+10:POKEV+21,PEEK(V+21)OR8:POKEDA
,1                                                     -052-
90 POKEV+16,PEEK(V+16)AND247:RETURN                    -180-
100 REM ********************************               -245-
110 BB=50902:FORI=1TO5                                 -035-
120 A=BB+80+140*RND(1):IFPEEK(A)<>32THEN120            -185-
130 B(I)=A:POKEA,201:NEXT                              -240-
140 GOSUB290:POKE683,.:POKEDA,.:POKEFA,1:AL=.:SYS4915
2:POKE679,.                                            -162-
150 IFRND(1)>.95ANDAL=.THENGOSUB570                    -155-
160 IFALTHENGOSUB20                                    -228-
170 BA=-(PEEK(BB)>OB)-(PEEK(BB+1)>OB)-(PEEK(BB+2)>OB)
-(PEEK(BB+3)>OB)                                       -171-
180 BA=BA-(PEEK(BB+4)>OB):A$=RIGHT$("[5 0]"+MID$(STR$
(INT(PU)),2),5):PU=PU+.02                              -169-
190 PRINT"{HOM}{ROF}{WHT}[6 ESP]PUNTOS: "A$"[4 ESP]BA
TERIAS:"BA                                             -105-
200 IFBA=5THEN580                                      -002-
210 IFPEEK(DA)=.THENIF(RND(1)>.6)ANDALTHENGOSUB50
                                                       -051-
220 IFPEEK(FA)=.THENON-((PEEK(678)AND9)=9)GOTO670:POK
EFA,1                                                  -006-
230 IFPEEK(678)=17ORPEEK(678)=9THENPOKEFA,.:GOTO670
                                                       -133-
240 IFPEEK(679)AND16THENGOSUB270                       -227-
250 IFPEEK(V+1)>194THENPOKEFA,.:GOTO670                -153-
260 GOTO150                                            -135-
270 PU=PU+10:POKE683,.:AL=.:POKEV+21,PEEK(V+21)AND207
:POKE679,.:RETURN                                      -036-
280 END                                                -145-
290 GOSUB1050:PRINT"{WHT}{CLR}";:POKE53280,.:POKE5328
1,.:POKE53265,PEEK(53265)AND239                        -240-
300 V=53248:BU=50643:DA=49969:GR=49734:FA=49739:XP=49
733:VE=49732:BO=VE+5                                   -221-
310 BB=BU+259:OB=200:SA=49970                          -137-
320 POKEV,160:POKEV+1,160:POKEV+4,100:POKEV+5,206:POK
EBB,32:POKEBB+1,32                                     -241-
330 POKEBB+2,32:POKEBB+3,32:POKEBB+4,32:POKEBB+255,32
:POKEBB+256,32                                         -155-
340 POKEBB+257,32:POKEBB+258,32:POKEBB+259,32
                                                       -085-
350 POKEV+28,17:POKEV+37,11:POKEV+38,14:POKEV+39,1:PO
KEV+40,7:POKEV+41,.                                    -059-
360 POKEV+42,7:POKEV+43,15:POKEV+44,.                  -193-
370 POKE2040,222:POKE2041,220:POKE2042,219:POKE2043,2
20:POKE2044,216                                        -068-
380 POKE2045,218:POKE53272,(PEEK(53272)AND241)OR14:PO
KEXP,0                                                 -187-
390 FORJ=.TO3:FORI=.TO39:POKE1744+I+40*J,PEEK(BU+I+PE
EK(XP)+256*J)                                          -244-
400 POKE56016+I+40*J,.:NEXT:NEXT:PRINT"{HOM}{10 ABJ}"
;                                                      -171-
410 PRINT"{GR2}[3 ESP]{RON}K@{ROF}[24 ESP]{RON}JCA{RO
F}[8 ESP]";                                            -090-
420 PRINT"[2 ESP]{RON}KGF@{ROF}[4 ESP]{RON}JCCA{ROF}[
13 ESP]{RON}JIGE @{ROF}[7 ESP]";                       -150-
430 PRINT" {RON}K[3 G]FB{ROF} {RON}JIGDE LL@{ROF}[8 E
SP]{RON}JI[3 G]F[2 ESP]@{ROF} {RON}KF@{ROF}[2 ESP]";
                                                       -245-
440 PRINT"{RON}K[4 G]E L[3 G]ED EE BA{ROF} {RON}KLL@K
[5 G]E[3 ESP]LGGEBA";                                  -117-
450 PRINT"{RON}[5 H][3 ESP][3 H] H[4 ESP]HH [3 H] [6
H][5 ESP]HH[3 ESP]";                                   -071-
460 POKEV+22,PEEK(V+22)AND247:POKEV+21,5               -115-
470 POKEVE,8:POKEBO,32:FORI=54272TOI+20:POKEI,.:NEXT
                                                       -029-
480 POKE54296,15:POKE685,1:A$=RIGHT$("[5 0]"+MID$(STR
$(PU),2),5)                                            -063-
490 PRINT"{HOM}{ROF}{WHT}[6 ESP]PUNTOS: "A$"[4 ESP]BA
TERIAS:"BA                                             -108-
500 PRINT"{HOM}{23 ABJ}{4 IZQ}<A>!*!<S>{ABJ}{3 IZQ}!-
! !-!{ABJ}{3 IZQ}<Z>!*!<X>";                           -193-
510 PRINT"{2 ARB}[28 ESP]LLEVANDO {HOM}"               -177-
520 PRINT"{HOM}{22 ABJ}[3 ESP]<R> <A>!*!<S> <A><R><S>
[3 ESP]<A><R><S> <A>!*!<S> <A><S><R>{14 DCH}";
                                                       -064-
530 PRINT"[3 ESP]!-! <Q><W>[3 ESP]!-![4 ESP]!3 -! <Q>
!*!<W> !-!<Z><W>{14 DCH}";                             -004-
540 PRINT" <Z>!*!<X> <Z>!*!<X>[2 ESP]<E>[4 ESP]<E> <E
> <E> <E> <E> <E>{HOM}"                                -104-
550 FORI=.TO15:POKE1064+350*RND(1),46:NEXT:POKE684,.
                                                       -089-
560 POKE53265,PEEK(53265)OR16:RETURN                   -184-
570 PA=.:GOSUB20:POKE683,1:AL=1:POKEV+21,PEEK(V+21)OR
48:POKE679,.:RETURN                                    -211-
580 POKEV+21,1:POKE54296,.:POKE683,.:POKE684,.:POKE20
40,222:POKEFA,.:POKEVE,8                               -102-
590 PRINT"{HOM}{2 ABJ}[9 ESP]BATERIAS CONECTADAS
                                                       -002-
600 PRINT"{ABJ}[6 ESP]EMISOR EN FUNCIONAMIENTO
                                                       -247-
610 PRINT"{ABJ}[11 ESP]AYUDA EN CAMINO                 -213-
620 POKE54283,.:POKE54283,33:POKE54296,15:PU=PU+50:A=
PEEK(GR):POKEGR,.                                      -193-
630 FORI=1TO200:POKE53280,4:POKEV+1,PEEK(V+1)+(PEEK(V
+1)>110):NEXT                                          -179-
640 POKE54283,.:POKE54296,.:FORI=1TO3000:NEXT
                                                       -205-
650 POKEGR,A:IFPU>RPTHENRP=PU                          -241-
660 GOTO740                                            -144-
670 POKEV+21,5:POKE683,.:FORI=1TO100                   -161-
680 POKEV+21,640/PEEK(V+21):POKE53280,1:NEXT:POKEV+21
,.:POKEVE,8:PU=.                                       -186-
690 PRINT"{HOM}{4 ABJ}{10 DCH}!HAS SIDO DESTRUIDO!
                                                       -039-
700 PRINT"{BLK}{11 ABJ}{9 DCH}DESEAS JUGAR MAS (S/N)?
{WHT}                                                  -138-
710 GETA$:IFA$="S"THEN740                              -201-
720 IFA$="N"THENPOKE890,.:SYS890                       -175-
730 GOTO710                                            -139-
740 FORI=.TO7                                          -048-
750 IFPEEK(B(I))=201THEN770                            -058-
760 POKEB(I),201                                       -135-
770 NEXT:GOTO140                                       -005-
780 POKE53280,.:POKE53281,.:PRINT"{CLR}[41 ESP]";
                                                       -009-
790 PRINT"[10 ESP]{WHT}SINDIN SOFT PRESENTA[10 ESP]";
                                                       -244-
800 PRINT"[40 ESP]";                                   -196-
810 PRINT"[3 ESP]{RON}{BLU}[3 ESP]{RED} {ROF} {RON}{B
LU}[3 ESP]{RED} {ROF} {RON}{BLU}[3 ESP]{RED} {ROF}[3
ESP]{RON}{BLU} {RED} {ROF}[2 ESP]{RON}{BLU} {RED} {RO
F}[2 ESP]{RON}{BLU}[2 ESP]{RED} {ROF}[2 ESP]{RON}{BLU
} ";                                                   -112-
820 PRINT"{RED} {ROF}{BLU} {RON} {RED} {ROF}{BLU}[2 E
SP]";                                                  -050-
830 PRINT"[3 ESP]{CYN}[4 ESP]{CYN}[6 ESP]{CYN}[7 ESP]
{CYN} {RON}{BLU} {RED} {BLU} {ROF}{CYN}[9 ESP]{CYN} {
RON}{BLU} {ROF} {RON} {RED} {ROF}{BLU}[2 ESP]";
                                                       -033-
840 PRINT"[4 ESP]{RON}{BLU} {RED} {ROF}[2 ESP]{RON}{B
LU}[2 ESP]{RED} {ROF}[3 ESP]{RON}{BLU} {RED} {ROF}[4
ESP]{RON}{BLU} {RED} {BLU} {RED} {BLU} {RED} {ROF} {R
ON}{BLU}[4 ESP]{RED} {ROF} {RON}{BLU} ";               -010-
850 PRINT"{RED} {BLU} {RED} {ROF}{CYN} {BLU}[2 ESP]";
                                                       -048-
860 PRINT" {RON}{BLU} {RED} {ROF} {RON}{BLU} {RED} {R
OF}[2 ESP]{CYN}[6 ESP]{RON}{BLU} {RED} {ROF}[4 ESP]{R
ON}{BLU} {RED} {ROF}[2 ESP]{RON}{BLU} {RED} {ROF} {RO
N}{BLU} {RED} {ROF} {RON}{BLU} ";                      -180-
870 PRINT"{RED} {ROF} {RON}{BLU} {RED} {ROF}{BLU} {RO
N} {RED} {ROF}{BLU}[2 ESP]";                           -118-
880 PRINT"[2 ESP]{RON}{BLU}[2 ESP]{RED} {ROF}[3 ESP]{
RON}{BLU}[3 ESP]{RED} {ROF}[2 ESP]{RON}{BLU} {RED} {R
OF}[4 ESP]{RON}{BLU} {RED} {ROF}[2 ESP]{RON}{BLU} {RE
D} {ROF} {RON}{BLU} {RED} {ROF} {RON}{BLU} {RED} ";
                                                       -083-
890 PRINT"{ROF} {RON}{BLU} {RED} {ROF}{BLU} {RON} {RE
D} {ROF}{BLU}[2 ESP]";                                 -060-
900 PRINT"[40 ESP]";                                   -197-
910 PRINT"[40 ESP]";                                   -198-
920 PRINT"{ACL}[40 ESP]";                              -097-
930 PRINT"[7 ESP]{WHT}!N!<Y>!M! <A><R><S><R> <R>[2 ES
P]<A><R><S> <R><A>!*!<S><A>!*!<S><A>!*!<S>[7 ESP]";
                                                       -015-
940 PRINT"[7 ESP]<H>C<N>[2 ESP]!-! !-! !-![2 ESP]!3 -
! !2 -![2 ESP]<Q><R><X>!-! !-![7 ESP]";                -038-
950 PRINT"[7 ESP]!M!<P>!N![2 ESP]<E> <Z>!*!<X>[2 ESP]
<E> <E> <E><Z>!*!<X><E><Z><X><Z>!*!<X>[7 ESP]";
                                                       -017-
960 PRINT"[40 ESP]";                                   -203-
970 PRINT"[40 ESP]";                                   -204-
980 PRINT"[40 ESP]";                                   -205-
990 PRINT"{GRN}[6 ESP]PULSA UNA TECLA PARA EMPEZAR[6
ESP]";                                                 -246-
1000 POKE198,.:WAIT198,1:POKE198,.                     -223-
1010 PRINT"{ABJ}{GRN}[8 ESP]DESEAS GRAVEDAD (S/N) ?"
                                                       -149-
1020 GETA$:IFA$="S"THENPOKE49734,1:GOTO100             -244-
1030 IFA$="N"THENPOKE49734,.:GOTO100                   -110-
1040 GOTO1020                                          -225-
1050 POKE53274,.:POKE788,49:POKE789,234:POKE56334,PEE
K(56334)OR1:POKE53281,.                                -160-
1060 RETURN                                            -199-
```





El travieso C-Byte tiene el honor de invitaros a la participación en nuestro tercer concurso de programación. Los requisitos necesarios son bien pocos:

- Saber programar un ordenador COMMODORE.
- Ser español o extranjero y
- Tener una edad comprendida entre 5 y 105 años.

Fácil, ¿verdad?
En cuanto a los premios, la mar de atractivos:

- **1.er premio.—60.000 pesetas en material informático a escoger por el premiado, más un póster de C-Byte.**
- **2.° premio.—30.000 pesetas en material informático a escoger por el premiado, más un póster de C-Byte.**
- **3.er premio.—15.000 pesetas en material informático a escoger por el premiado, más un póster de C-Byte.**

Y en fin, si alguno de los programas destaca por su originalidad, estética o comicidad, no sería de extrañar que le cayera alguna cosilla más...

BASES DEL CONCURSO

1. Los programas remitidos al concurso deberán ser creación original del autor o autores, y completamente inéditos, pudiendo remitir tantos programas como se desee.

2. Los programas deberán ser enviados en casete o disco flexible a TU MICRO COMMODORE (Concurso de programación). Apartado de Correos 61.294. 28080 MADRID.

3. Los programas podrán ser de cualquier tipo (juegos, utilidades, gestión, educativos) y habrán de estar escritos en lenguaje BASIC o Código Máquina.

4. Los programas deberán ser remitidos desprovistos de cualquier tipo de protección, que impida o dificulte el análisis del mismo, así como reproducción en estas páginas y su introducción como listado siguiendo el sistema FUERA ERRORES.

5. Cuando la ejecución del programa precise de la concurrencia de determinado periférico o aditamento (joysticks, tabletas gráficas, programas comerciales de ayuda), se valorará la indicación de las modificaciones pertinentes, para que el programa pueda ser disfrutado por cualquier usuario en la configuración básica.

6. Todo programa presentado al concurso deberá acompañarse de los siguientes datos:

- Datos personales del concursante.
- Nombre del programa.
- Modelo para el que está destinado.
- Breve descripción del programa detallando las indicaciones necesarias para su ejecución.

7. Los programas premiados pasarán a ser propiedad de la revista TU MICRO COMMODORE, pudiendo hacer ésta libre uso de ellos, y renunciando sus autores a cualquier otra compensación distinta al premio.

8. Los programas no premiados que por su calidad se hagan merecedores de su publicación, serán adquiridos por la editorial, aplicando la tarifa vigente.

9. Los programas recibidos con posterioridad a la fecha tope de la presente edición, serán automáticamente incluidos en los destinados a la siguiente.

10. El jurado decidirá sobre todos los aspectos no contemplados en estas bases y su decisión será inapelable.

Y ahora a darse mucha prisa, el plazo para la recepción de programas termina el próximo día 1 de septiembre.

¡¡SUERTE!!



Tour de France

Para aquel que guste de emular sobre la bicicleta a los grandes ases de las dos ruedas, y no esté especialmente dotado para tal fin porque los compañeros le dejan atrás a las primeras de cambio, y la más mínima subida le cuesta sangre, sudor y lágrimas, presentamos otro gran entretenimiento para Commodore-64, el Tour de France (léase «tur de franss»), con el que podrá retar a sus competidores y dejarles en ridículo sentado en su sillón.

Antes de empezar, deberemos seleccionar el nombre y nacionalidad de los seis representantes en este evento deportivo, y después, si queremos correr el total del Tour (Grand Circuit) o por el contrario elegir un número determinado de etapas (Choisir Etapes). Si nos hemos inclinado por la última opción, aparecerá ante nuestros ojos el mapa de Francia, con el recorrido total que podemos llevar a cabo, debiendo elegir las etapas que más nos atraigan, en el número que prefiramos.

A continuación, y antes de empezar a correr, deberemos darle colores a los «coulottes» (palabra francesa de obvia interpretación) y maillots de cada corredor; y ya estamos preparados para afrontar esta nueva edición de la carrera más importante que existe sobre la bicicleta.

Al aparecer en la línea de salida, nuestra concentración se halla en su punto más álgido, y ya casi no oímos el rugir del público, que se agolpa a los costados de la carretera, y a lo largo de todo el circuito, animando sin cesar a todos los corredores. El director de la carrera da la señal de salida y es entonces cuando, al principio trabajosamente, y después ya con una mayor facilidad, vamos adelantando terreno moviendo los desarrollos de nuestra bicicleta e impulsándonos además de con los pedales, con la fuerza de nuestro cuerpo.

Poseemos, como quien dice, dos diferentes marchas, que podemos aplicar como cambios de piñón, los cuales deberemos usar adecuadamente dependiendo de que el trazado sea llano, en pendiente o en bajada. A lo largo del recorrido podremos ver determinadas señales que nos indicarán la inclinación del recorrido. Mucho cuidado con los márgenes de la carretera, pues si nos salimos de ellos colisionaremos, yendo a parar con nuestros huesos al suelo. El tiempo de cada etapa se irá contabilizando individualmente cara la clasificación general, de modo que corramos lo más rápido posible para enfundarnos el maillot amarillo de ganador.

PUNTUACION

Nombre: Tour de France

Precio: 2.200 ptas

Soporte: Cinta

Ordenador: Commodore 64 ó 128

VEREDICTO FINAL

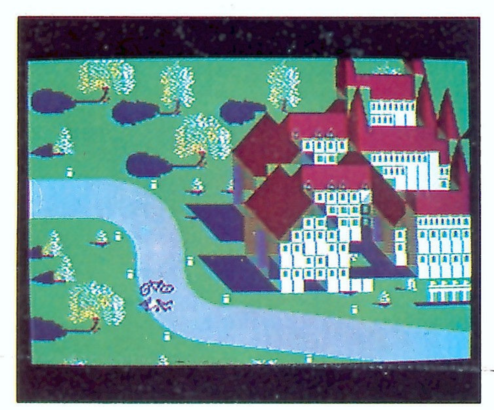
Gráficos: * * * * *

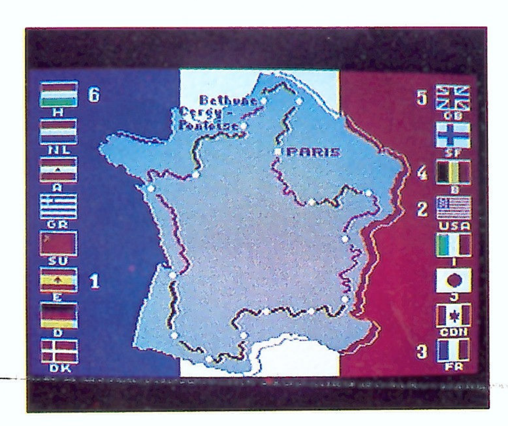
Sonido: * * * *

Originalidad: * * * * *

Dificultad: * * * *

Interés: * * * *









Kung-Fu Master

Después de pasar largos años aprendiendo la técnica y dominio de las artes marciales, y siendo un experto en Kung-Fu, te encaminas hacia la que va a ser una de tus mayores pruebas, ya que tienes que entrar en el castillo del mago y rescatar a su joven cautiva.

Lo malo es que este templo está custodiado por multitud de guardianes, a cual más feroz, dragones, murciélagos y otros extraños personajes, como enanos, terribles lanzadores de cuchillos, gigantes y expertos luchadores. Para llegar a salvar a esa muchacha, deberás pasar por cinco diferentes plantas, cada una de las cuales está custodiada por diferentes seres, a los cuales deberás ir quitando de tu camino, y llegar al extremo del corredor donde se hallan las escaleras, a través de las cuales podrás acceder al siguiente nivel.

Tus enemigos serán los gerrilleros, presentes en todos los niveles, pesados luchadores que se te aparecerán desde cualquier lado, casi siempre en grupo e intentando agarrarte hasta lograr quitarte toda la energía, con una patada o un simple puñetazo te librarás de ellos; los tiradores de cuchillos son más peligrosos, ya que dos impactos de sus certeros proyectiles te mandarán al otro mundo, y su resistencia a tus golpes es mayor.

En la segunda planta te encontrarás con serpientes, que aparecen al romperse jarrones que caen del techo, dragones que surgen de balones y te lanza sus llamas y peligrosos globos que explotan en mil pedazos; también te encontrarás por vez primera con los gnomos, seres muy hábiles y rápidos, difíciles de golpear por su reducida estatura; si logras llegar a otras plantas te atacarán las avispas, que te perseguirán hasta que consigan picarte y los enanos y guerrilleros se aliarán, así como los lanzadores de cuchillos, que en ocasiones te atacarán al unísono.

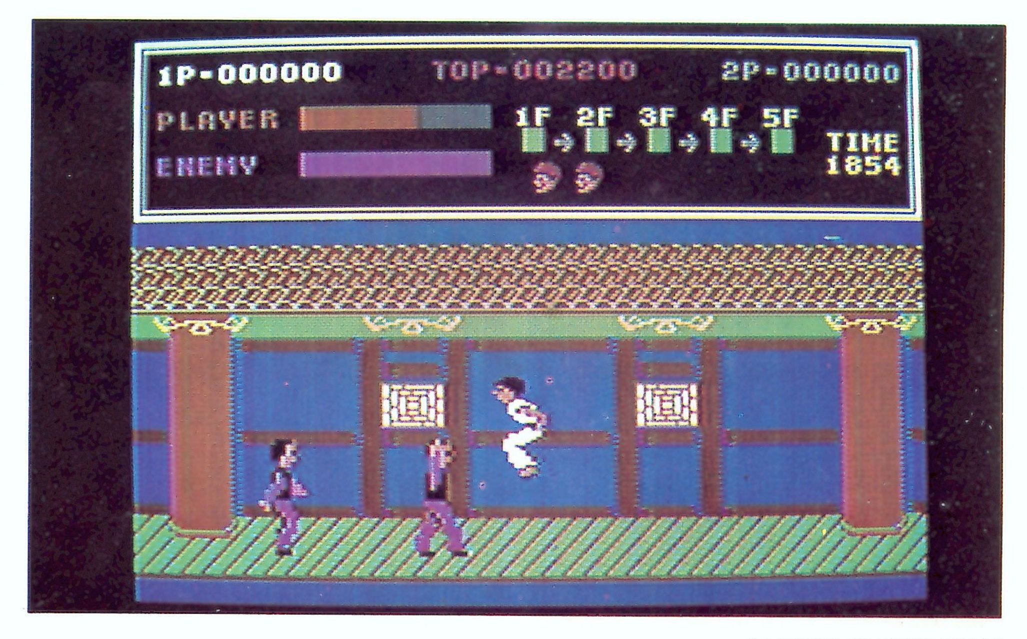
Además, al final de cada nivel, e inmediatamente antes de llegar a las escaleras que conducen al siguiente, se encuentra un guardián, los cuales están muy preparados y serán en muchos aspectos muy superiores a tí. En la primera planta se encuentra el experto en numchacos, los cuales maneja rápida y precisamente; en el segundo nivel se haya un lanzador de estrellas cortantes, que vuelven a su mano como si fuera un boomerang; el hombre gigante es el siguiente guardián, y éste te saldrá al paso con las manos desnudas, pero un solo golpe suyo acabará contigo; los demás serán lanzadores de hechizos, y en la última un hábil luchador de Kung-Fu, mucho más rápido, resistente y fuerte que tú, ¡te lo pondrá muy difícil!

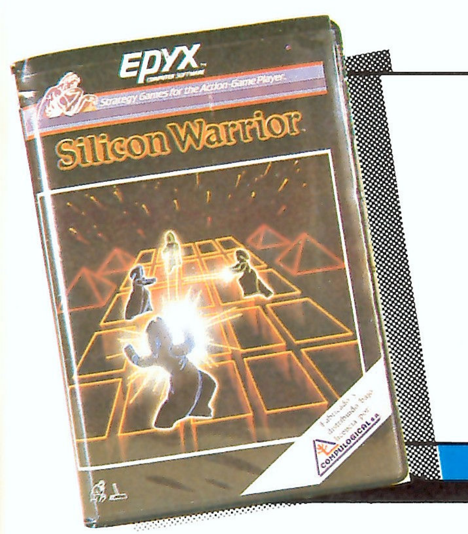
PUNTUACION

Nombre: Kung Fu Master
Precio: 2.600 ptas
Soporte: Cinta
Ordenador: Commodore 64

VEREDICTO FINAL

Gráficos: * * * *
Sonido: * * *
Originalidad: * * *
Dificultad: * * * * *
Interés: * * * * *





Silicon Warrior

Silicon Valley se ha convertido, en el año en curso, en la zona más tecnificada de todo el planeta, carrera científica que empezó en los lejanos años 80, hasta llegar a hoy, año 2084.

PUNTUACION

Nombre: Silicon Warrior
Precio: 2.300 ptas.
Soporte: Cinta
Ordenador: Commodore 64 o 128

VEREDICTO FINAL

Gráficos: * * * *
Sonido: * * *
Originalidad: * * *
Dificultad: * * * *
Interés: * * * *

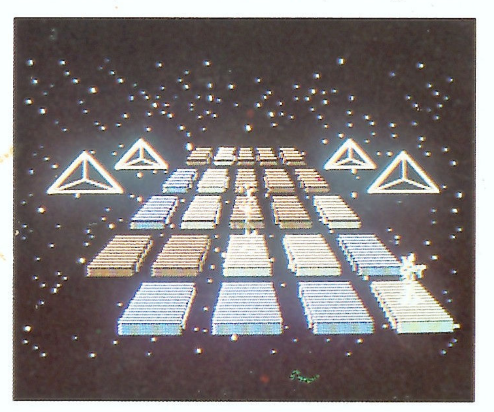
Una raza superhumana de tecnócratas ha inventado las computadoras de la décima generación, con una RAM superior a todas las anteriores y una inteligencia artificial capaz de asombrar al mas cualificado en la materia. Esta raza, los Syborgs, ha decidido enseñar a construir una computadora similar, a la casa terráquea que demuestre una mayor agresividad tecnológica. Por ello, las casas más pujantes de la Tierra enviaron al campo futurista en que los Syborgs decidieron sería la batalla, a sus guerreros tecnológicos más importantes, de los cuales únicamente sobreviviría uno para llevarse la patente de vuelta a nuestro mundo.

Nosotros somos uno de esos guerreros escogidos, para una lucha que no será con las armas, sino con la inteligencia y la rapidez de pensamiento, ya que deberemos programar en un campo estelar de cinco chips de largo por cinco de ancho, el mismo número de estos, así haciendo un fácil cálculo de posibilidades, gozaremos para ello de todas las diagonales y horizontales, así como de las diagonales mayores, únicas que contienen cinco chips.

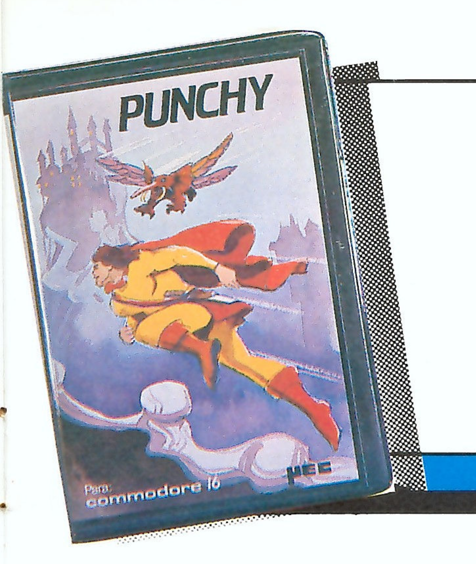
Logrando programar estos cinco habremos conseguido descubrir el código que revelará todos los secretos de la computadora de la décima generación. Lo esencial será que esos secretos, a ser posible caigan en buenas manos, y por ello habrá que tener cuidado con los guerreros Warriors, terribles enemigos que codician el secreto de la computadora para usarlos con perversos fines y llegar a dominar mediante la guerra al resto de los planetas del universo. Lucharán con malas artes obstaculizando al adversario y cambiando sus chips para conseguirlos en su provecho y proclamar que son suyos, además de utilizar su láser para agotar las energías de su oponente o empujarle hacia el agujero negro que haya aparecido entre los chips.

El modo de programar los cinco chips será teletransportándose de uno a otro, puesto que se hayan flotando en el vacío, y mutarlos al código del color elegido, si esto es llevado a cabo empezarán a lanzar destellos, indicadores de nuestra victoria, pero no nos creamos vencedores demasiado pronto, ya que no se nos revelará el secreto de la computadora hasta haber conseguido derrotar a nuestros competidores en cinco difíciles batallas.

Para defendernos de los ataques de los enemigos, podremos utilizar nuestros láseres, con el objetivo de debilitarles y consiguientemente, obligarles a retornar a la pirámide de energía, ganando un tiempo precioso para acabar nuestra misión.



Si queremos ganar la batalla de Silicon Valley, tendremos que tener una previa estrategia en nuestro cerebro, entre ellas la más usada es la de atacar ferozmente cubriendo los flancos en que tenemos programado algún chip para impedir que el enemigo nos cambie el color; otra estrategia es la de estorbo, en la que a medida que nuestro oponente vaya programando chips, iremos teletransportándonos de uno a otro. Cambiándolos a nuestro código de color; una tercera sería la de comprobar la empleada por el adversario, para conocer sus fallos y lograr vencerle más fácilmente. Sobre todo, sé prudente, rápido e inteligente, porque recuerda que sólo el más apto sobrevivirá.



Punchy

De nuevo tomamos interés por los sufridos usuarios del hermano pequeño de nuestro Commodore-64 y para que no digan nada las malas lenguas, incluiremos en nuestros SOFTMODORES periódicamente algún que otro comentario sobre un juego del Commodore-16.

PUNTUACION

Nombre: Punchy
Precio: 1.500 ptas.
Soporte: Cinta
Ordenador: Commodore 16

VEREDICTO FINAL

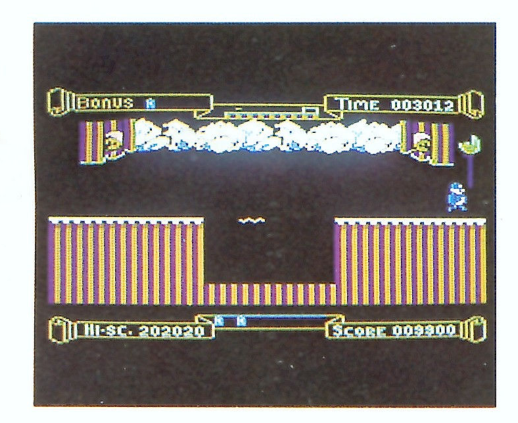
Gráficos: * * *
Sonido: * *
Originalidad: * * *
Dificultad: * * * *
Interés: * * * *

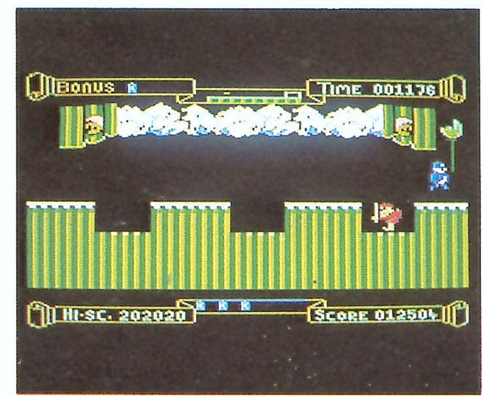
Las historias de chico-salva-chica, o la consabida de caballero medieval salva princesa que está apresada por terrible dragón, vuelven a nosotros con Punchy, que dentro de la limitada memoria del Commodore-16 aparece como un juego, porqué no decirlo, bastante conseguido, tanto en animación y entretenimiento, como en gráficos y presentación.

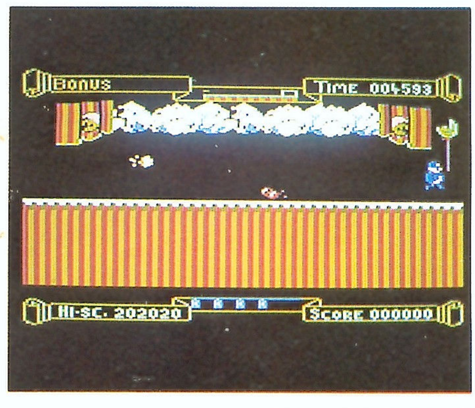
En el caso que tenemos entre manos, el malvado Punch (porque siempre debe haber un malvado para propiciar rescates peligrosos y zarandajas por el estilo) ha secuestrado a Judy (las intenciones que guarda al respecto las callamos), y Bobby, el estupendo, fornido, guapo, valiente y resto de cualidades que se le asignan al bueno de la película, debe rescatarla, para lo cual, por supuesto, va a tener que pasar por multitud de peligros variados y muy distintos de los que tiene que superar un policía en su trabajo normal (suponemos que Bobby pedirá un sobresueldo por meterse a salvador de chicas secuestradas).

A lo largo de su camino, que va a estar dividido en varias pantallas, en cada una de las cuales va creciendo la dificultad sucesivamente, Bobby va a tener que esquivar tartas de crema voladoras, tomates podridos, saltar agujeros gigantes que amenazan con tragárselo el menor descuido, e incluso puede que se encuentre con el mismísimo Punch, esperpéntico jorobado que le perseguirá e intentará, como no, acabar con él.

También se encontrará en su recorrido, ríos y precipicios que atravesar, donde la única ayuda será la pericia en saltar a una especie de almadia (véase balsa), la cual va de un lado a otro, impulsada por la corriente (de aire o de agua, depende del caso en particular), e incluso, al ir a superar los terribles agujeros que existen en el suelo deberá tener cuidado, pues puede ser pinchado de forma muy molesta en pleno salto por algún personaje escondido en la negrura del hoyo. De todos modos, creemos que con tu ayuda, nuestro simpático Bobby podrá llevar a buen fin su aventura, a lo mejor de lo contrario le bajan de puesto en el cuerpo de policía.





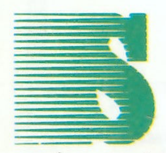




Back to the Future

Los problemas y oportunidades que representaría al hombre poder volver al pasado serían de diversa índole. Quizás la más significativa consistiría en poder cambiar el curso de la historia, aunque hoy por hoy no sabemos si sería posible, ya que se podrían plantear multitud de paradojas, como la que nos ocupa en «Back to the Future».



Supón que un muchacho, por un azar del destino, retrocediera en el tiempo aproximadamente unos treinta años, y se encontrará con su futura madre, una jovencita veinteañera, y su también futuro padre. El problema no sería tal si no ocurriera que su madre, Lorraine Barnes, se enamorara de él, su hijo, no hacienco ningún caso del que será su marido, George McFly. Por ello, Marty debe poner las cosas en su debido lugar antes de poder regresar al futuro, o sea, tendrá que desembarazarse de su madre (un poco rollo la señora), y hacer que ésta se fije, y se enamore de George, con el objeto de que su futuro exista (¿todo un problema, no?).

Marty es ese muchacho, el cual siempre se retrasaba a la hora de acudir a algún sitio, pero desde que se sentó al volante de un Delorean con propulsión nuclear inventado por el profesor Brown, nunca volverá a hacerlo, ya que va a regresar treinta años atrás, y hasta que no cumpla su objetivo, no podrá retornar.

Los cinco personajes que componen esta historia son el propio Marty; el Doctor Emmet Brown, constructor de la máquina del tiempo; Biff Tannen, el matón del colegio, que entorpecerá en todo momento la labor de Marty, golpeándole y dejándole «groggy» en el suelo; Lorraine Barnes, futura madre de Marty, que si no se casa con George, provocará que su familia deje de existir; y George McFly, padre de nuestro héroe, tímido y apocado, que se aparta de la acción.

De este modo, Marty debe conseguir que George y Lorraine estén lo más juntos posible, y al mismo tiempo, que ésta se le acerque lo menos posible, ya que cuanto más rato esté junto a él, su futuro y el de su familia, representados en el juego por una fotografía de esta y otra de Marty, se irán difuminando cada vez más hasta desaparecer, perdiendo así el protagonista toda oportunidad de nacer, y consecuentemente, de volver al futuro.

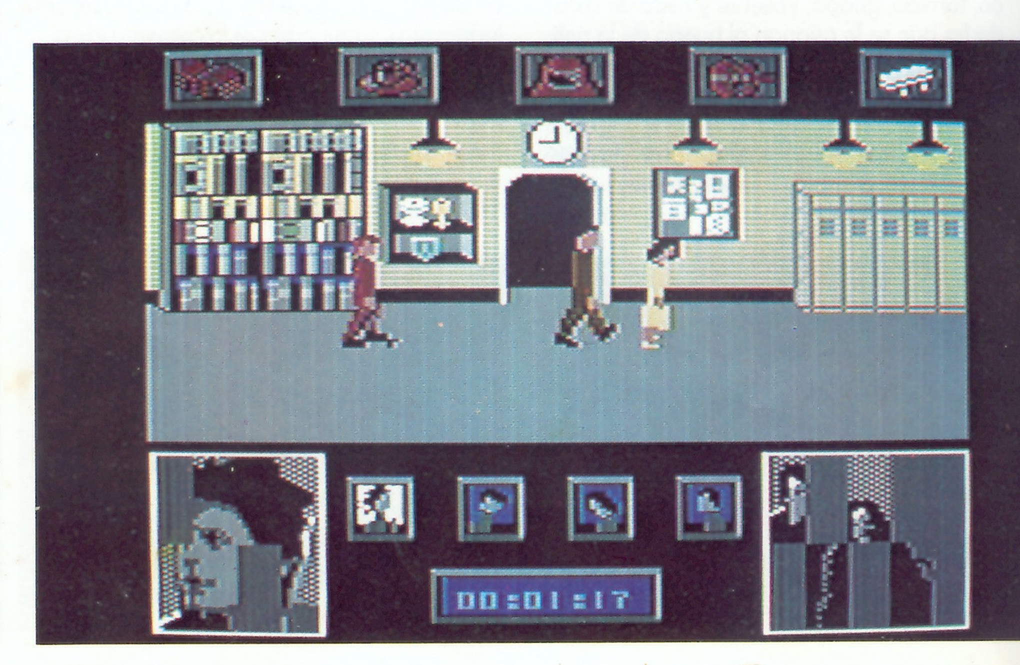
Así que, ingéniatelas para, utilizando diversos objetos que irás encontrando en tu camino y que influirán en cada personaje de un determinado modo, conseguir tu objetivo y retornar a tu tiempo, con la satisfacción de haber hecho de «Celestino» de tus propios padres.

PUNTUACION

Nombre: Back to the Future
Precio: 2.300 ptas.
Soporte: Cinta
Ordenador: Commodore 64 ó 128

VEREDICTO FINAL

Gráficos: * * * *
Sonido: * * *
Interés: * * *
Dificultad: * * * * *
Originalidad: * * * * *



CURSO DE BASIC

FUNCIONES NUMÉRICAS

Hemos hablado ya de algunas de las funciones de que dispone el vocabulario BASIC, concretamente, de las aplicaciones al tratamiento de expresiones y variables de cadena (string). Estas son: **LEN, STR$** y **VAL.**

Existe además, otro tipo de funciones de notable importancia, aplicables de manera general a expresiones de tipo numérico a las que denominamos FUNCIONES NUMERICAS, cuyo funcionamiento y misión trataremos en adelante. El formato general de estas funciones es similar al de las ya descritas de cadena. Así pues, constan de función y argumento (constante, variable o expresión de tipo numérico) que será evaluado para obtener el resultado.

Podemos decir, por tanto, que la sintaxis general de cualquier función BASIC sea del tipo que sea es:

Función (argumento).

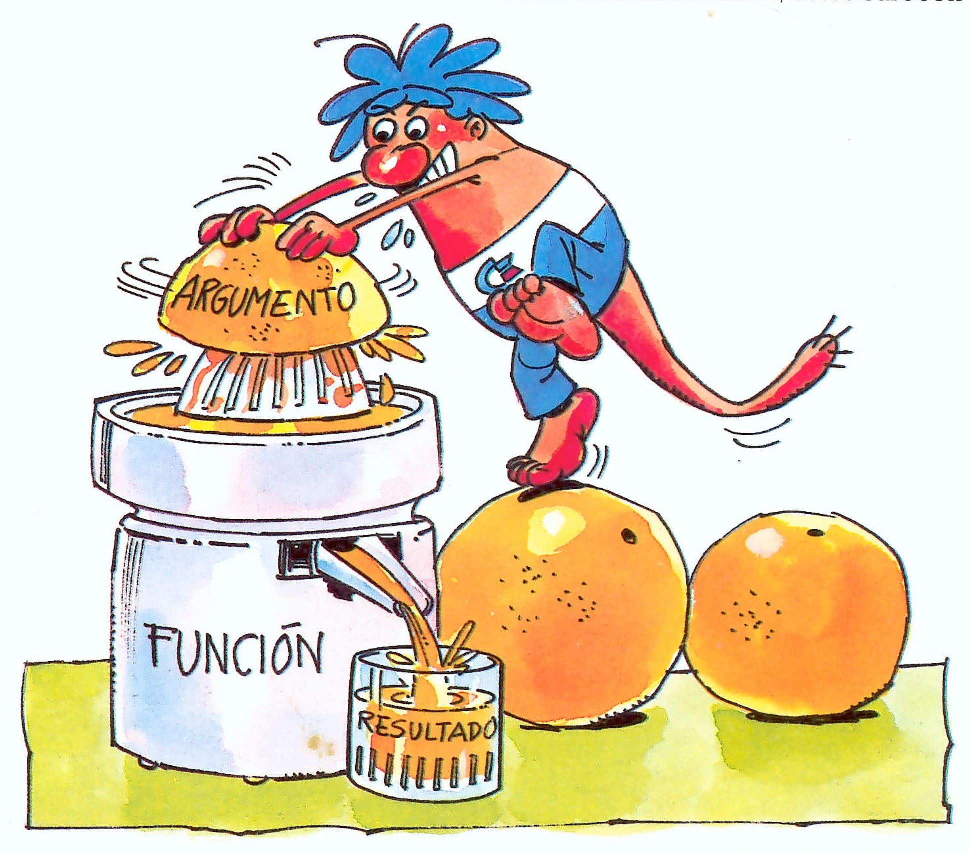
Donde el argumento puede ser una constante, una variable o una expresión a evaluar; siempre y cuando éste se ajuste al tipo de datos sobre los que opera la función: alfanuméricos o numéricos. Podemos establecer tres grupos dentro de las funciones numéricas BASIC: las matemáticas comunes, las logarítmicas y las trigonométricas.

Entre las funciones matemáticas se cuentan: **SGN, ABS, INT, SQR** y **RND;** entre las logarítmicas: **LN** y **EXP** y entre las trigonométricas: **SIN, COS, TAN, ASN, ACS** y **ATN.** Además, de todas estas funciones hablaremos de algunos otros elementos BASIC relacionados con ellas.

LA FUNCIÓN SGN

Por medio de esta función es posible determinar el signo de un argumento. Evidentemente, **SGN** no es aplicable a variables de cadena, y sí en cambio a constantes, variables o expresiones de tipo numérico.

Tengamos en cuenta que aunque el primer carácter de una cadena sea un espacio en blanco = o un signo menos —, a efectos del ordenador, por encontrarse entrecomillados, estos carecen



Las funciones numéricas constan de función y argumento, que será evaluado para obtener el resultado.



Podemos establecer tres grupos dentro de las funciones numéricas BASIC: las matemáticas comunes, las logarítmicas, y las trigonométricas.

de su función habitual de signo matemático. Para ser más exactos, el espacio en blanco o el signo menos – al principio de una cadena, no son otra cosa que simples caracteres.

Conviene recordar ahora la existencia de la función **VAL** cuya misión es convertir una variable de cadena en numérica. Gracias a ella podemos someter una variable alfanumérica a un proceso previo de conversión, tras el cual poderle aplicar la función **SGN.**

Supongamos que queremos saber cuál es el signo de la constante menos tres **–3.** En este caso, por tratarse de un valor numérico, podemos aplicar directamente la función **SGN** de la siguiente manera:

```
PRINT SGN(–3)
```

Si por el contrario, deseáramos conocer el signo del literal menos 3 **"–3",** deberíamos recurrir primero a la función **VAL:**

```
PRINT SGN (VAL("–3"))
```

Cualquier intento de evaluar el signo de una cadena sin pasar previamente por la función **VAL,** será rechazado sin miramientos con el error **TYPE MISMATCH ERROR.** Puede que no hallemos mucha utilidad en aplicar una función sobre una constante: es evidente que no nos hace falta para conocer el signo de menos 3 **(–3).**

Conviene por tanto, recordar el hecho de que las constantes no tienen por que ser tan simples como la que hemos empleado en el ejemplo. Para reafirmar este hecho transcribimos una constante cuya complejidad habla por si sola:

```
(–3*5+SQR(15*3/2)–28*5/3)*2–66
```

En los casos en que no entren en juego variables es conveniente que en vez de introducir en el programa el cálculo de la función, hallemos su valor mediante el modo directo para insertar este en el programa. De esta forma ahorraremos memoria y tiempo de ejecución, puesto que la expresión completa siempre ocupa más espacio que su resultado, y además tiene que ser recalculada cada vez que el ordenador pasa por la línea concreta de programa; operación bastante inútil si tenemos en cuenta que su valor es constante.

La función **SGN** indica el signo de su argumento pero de forma diferente a como en principio podíamos pensar, es decir, con un signo más **(+)** o un signo **(–).** Entre otras cosas porque estos caracteres no son numéricos y por tanto no pueden ser resultado de una función de este tipo.



Entre las funciones matemáticas se cuentan: ***SGN, ABS, INT, SQR*** *y* ***RND;*** *entre las logarítmicas:* ***LN*** *y* ***EXP;*** *y entre las trigonométricas:* ***SIN, COS, TAN, ASN, ACS*** *y* ***ATN.***

*Por medio de la función **SGN** es posible determinar el signo de su argumento.*

Las posibles respuestas a la ejecución de la función **SGN** son: 1 si el argumento resulta ser positivo. 0 si el valor es nulo y −1 si éste es negativo. Veamos algunos ejemplos:

```
10 REM — MANEJO DE LA
   FUNCION SGN
20 A=−2:B=0:C=3
30 PRINT SGN(A), SGN(B),
   SGN(C), SGN(A/B), SGN(A/C),
   SGN(C/A)
```

La utilidad de esta función puede que todavía no se nos haga evidente, pero según vayamos avanzando en conocimientos tomaremos conciencia del servicio que puede prestar. No olvidemos que algunas operaciones matemáticas, incluso de uso muy común, como la radicación, no admiten valores negativos. Por ejemplo, si tenemos una instrucción del tipo:

```
10 PRINT SQR(A)
```

Podemos correr el riesgo de que la variable **A** adopte un valor negativo. De ser así, al ejecutarse la línea 10 el programa se detendrá con un mensaje de error del tipo **ILLEGAL QUANTITY ERROR.** Una forma de evitar esto es utilizar la función **SGN** para no dejar llegar a la operación nada más que valores positivos o cero:

```
10 IF SGN(A))=0 THEN PRINT
   SQR(A)
```

En el caso concreto del ejemplo, hubiera sido más sencillo emplear directamente la comparación sobre **A:**

```
10 IF A>= THEN PRINT SQR(A)
```

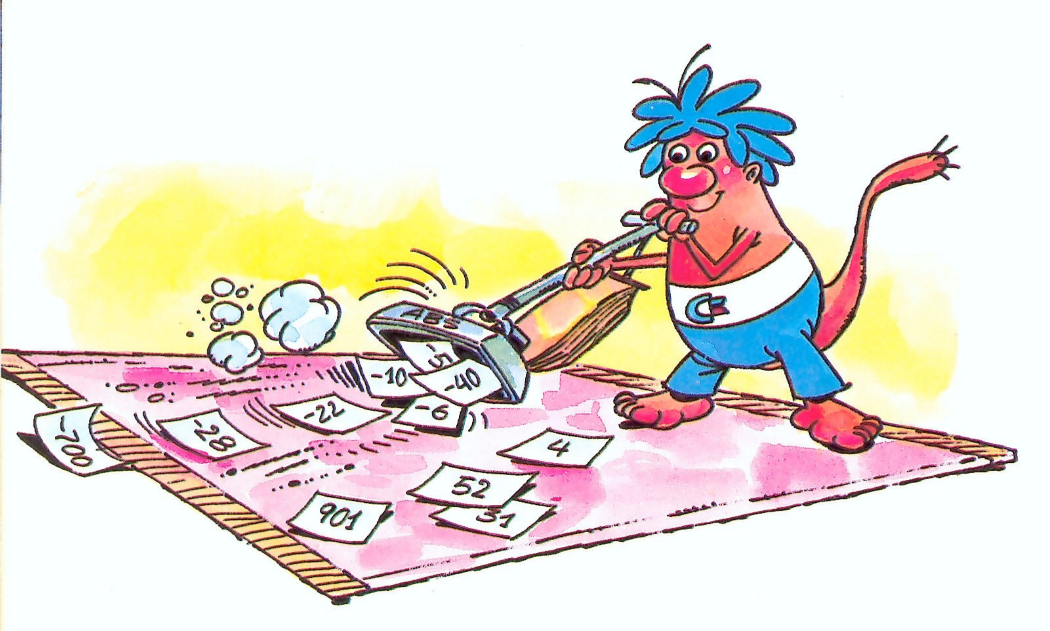
Aunque con **SGN** el programa gana en claridad de las intenciones del programador sobre esa línea de texto BASIC. En algunas ocasiones no es sólo conveniente o clarificador, sino inevitable recurrir a **SGN.**

LA FUNCIÓN ABS

Esta función permite obtener el valor absoluto de un número. Matemáticamente, entendemos por obtener el valor absoluto, privar de su signo al número afectado. El valor absoluto de un número se representa convencionalmente entre barras verticales. Para



*5. Gracias a la función **VAL** podemos someter una variable alfanumérica a un proceso previo de conversión, tras el cual poder aplicar **SGN.***



__ABS__ permite obtener el valor absoluto de un número.

ser más exactos, la obtención del valor absoluto más que desproveer del signo al argumento, lo que hace es forzarlo a ser positivo, cualquiera que fuera su signo anterior.

De lo dicho se desprende que al aplicar la función **SGN** sobre un valor absoluto, nunca obtendremos el valor −1. La utilidad de esta función es grande, pongamos un ejemplo de ello:

Vamos a suponer que en el diseño de un programa de juego en el que se intenta alcanzar con el lanzamiento de un proyectil a un tanque en movimiento, deseamos indicar al usuario lo alejado que quedó el último disparo de su objetivo, con el fin de corregir la trayectoria en la próxima andanada.

Llegados a este punto veremos que se pueden producir dos circunstancias: o bien el disparo quedó largo o bien corto (desecharemos las posibilidades de izquierda y derecha para no complicar el ejemplo).

En la variable **D** tendremos el valor de la posición del tanque menos la de impacto del proyectil. Si suponemos que el origen de coordenadas se encuentra en nuestra base de lanzamiento artillero, cuando la variable **D** resulte positiva el disparo habrá sido corto: de ser negativo largo, y siendo cero un éxito completo. Pasemos ahora a la transcripción efectiva del planteamiento a un programa BASIC:

```
10 ...
20 IF D<>Ø THEN 50
30 PRINT "** FELICIDADES
   **":PRINT
40 PRINT "IMPACTO
   DIRECTO":GOTO 90
50 PRINT "EL DISPARO HA
   QUEDADO ";
60 IF SGN(D)=1 THEN PRINT
   "CORTO";:GOTO 80
70 PRINT "LARGO";
80 PRINT " POR" ABS(D)
   "METROS"
90 ...
```

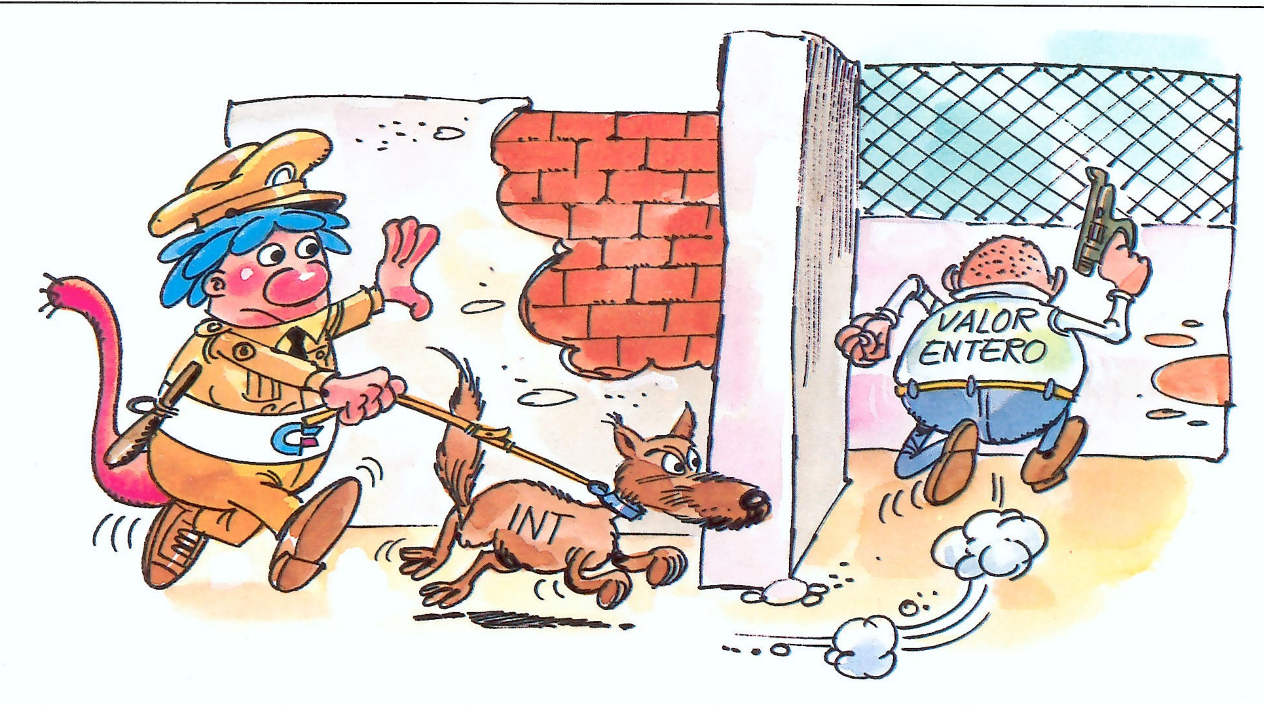
Con esta sencilla rutina se imprimirá en pantalla indicando el resultado del disparo. Indudablemente, habremos mejorado la presentación de datos del programa imprimiendo, por ejemplo:

```
EL DISPARO HA QUEDADO LARGO
POR 15 METROS
```

que no presentando la variable **D** en su valor con signo (sin utilizar **ABS**), como sería el caso de:



El valor absoluto de un número se representa convencionalmente entre barras verticales.



*La misión de **INT** es encontrar el valor entero del argumento numérico específico.*

EL DISPARO HA QUEDADO A –15 METROS

La rutina del ejemplo nos da pie para comentar algunas características de la programación, no relacionadas concretamente con ciertas sentencias BASIC, sino más bien con el diseño general de cualquier programa. Por una parte, como ya hemos visto, podíamos haber evitado la inclusión de las funciones **SGN** en las líneas 60 y 70, si bien es cierto que con un ahorro de memoria y tiempo de ejecución bastante insignificantes.

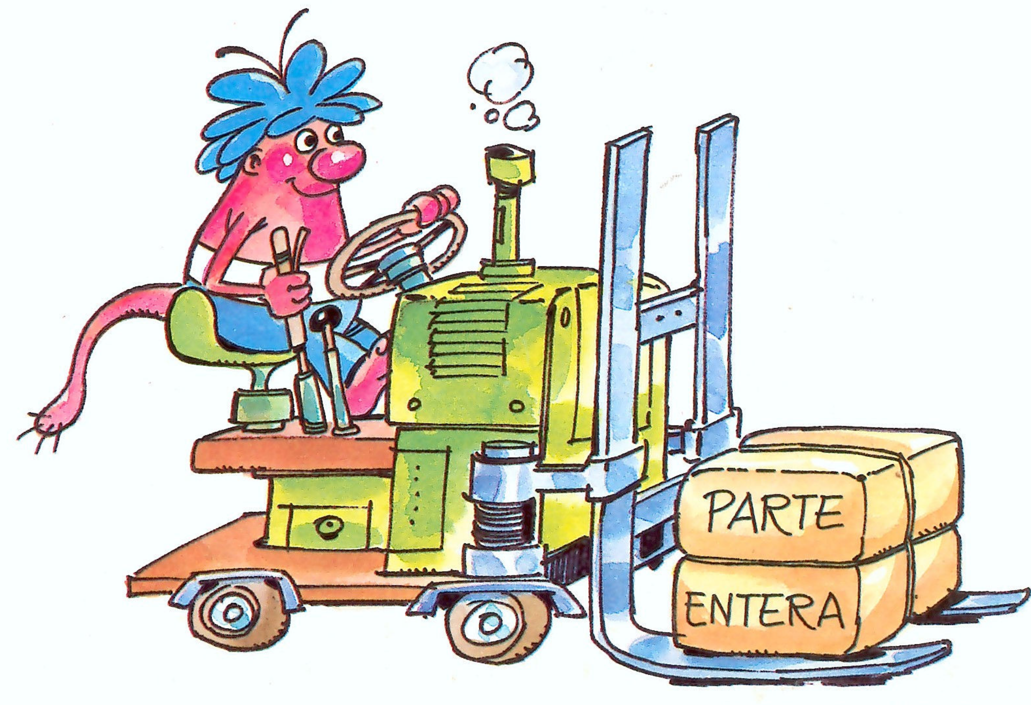
De mayor importancia es la forma de disponer las sentencias **IF** de las líneas 20 y 60, puesto que este tipo de instrucciones de decisión retardan bastante la ejecución de un programa. Como en el caso de las funciones **SGN,** la pérdida de tiempo por evaluar los dos **IF** carece de importancia en este caso, aunque es una buena costumbre ahorrar memoria y tiempo de ejecución en todos y cada uno de los elementos de la programación, si pensamos que en la totalidad de un programa de relativa longitud, la diferencia de ese conjunto de pequeños ahorros si se hace sentir.

En la rutina propuesta nos encontramos en el caso de dar diferente tratamiento a tres posibles circunstancias. La primera de ellas se produce cuando el valor de la variable **D** resulta ser nulo, caso este en que se ejecutarán las líneas 30 y 40, antes de continuar la ejecución en la línea 90.

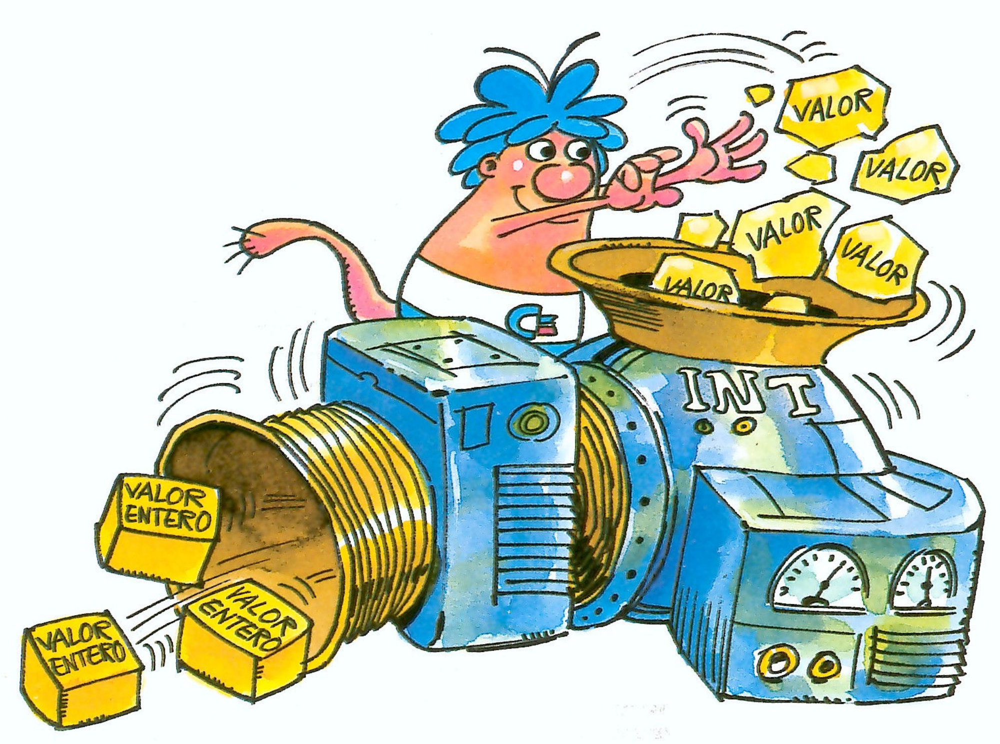
En caso contrario se ejecuta siempre la línea 50, para establecer una nueva comparación encaminada a determinar si el valor de **D** es positivo o negativo, con el **IF** de la línea 60. En base a este resultado se imprime directamente el mensaje de LARGO o CORTO, por ser estos **IF** exclusivos, es decir, si **D** no es cero ni positivo, por ejemplo, será forzosamente negativo.

LA FUNCIÓN INT

Entre las funciones BASIC dedicadas al tratamiento numérico que no constituyen una operación matemática, es probablemente **INT** la de uso más frecuente. La misión de esta función es encontrar el valor entero del argumento especificado, ya sea éste entero o decimal, positivo o negativo, despreciando en cualquier caso la parte decinal si la hubiera.



*Cuando **INT** se aplica sobre una variable no altera su contenido, simplemente devuelve su valor entero.*



*Una de las aplicaciones de **INT** puede ser la entrada de datos, para adaptar al formato entero los valores introducidos.*

Aunque el BASIC considera como habitual el empleo de valores numéricos con coma flotante (con decimales), gran cantidad de sucesos representables por números y que puedan tratarse en un programa resultan ser valores enteros. A modo de ejemplo podemos citar los números de teléfono, los valores que podemos obtener al lanzar uno o dos dados, etc.

Debido a esta circunstancia resulta interesante disponer de la posibilidad de cerciorarnos que el valor sometido a determinado tratamiento, una representación en pantalla por ejemplo, es entero y no contiene decimales. Como el resto de las funciones, **INT** no modifica su argumento, limitándose a obtener su parte entera para el tratamiento directo o almacenamiento en otra variable.

Así mismo, tampoco es de importancia el hecho de que el argumento de la función sea ya entero, puesto que aunque aparentemente no se produzca efecto alguno, la tarea de conversión se lleva a cabo sin producirse ninguna detención por error. Una de las aplicaciones de la función **INT** puede ser en las entradas de datos, para adaptar al formato entero los valores introducidos.

Supongamos, por ejemplo, que durante la creación de un programa de juego interrogamos al operador sobre el nivel de dificultad que desea escoger. Una depuración inmediata es la del rango máximo que puede admitirse para este dato, por ejemplo entre uno y tres. Observemos el siguiente programa:

```
10 INPUT "NIVEL DE
   DIFICULTAD";N
20 IF N<1 OR N>3 THEN 10
30 IF N=1 THEN...
40 IF N=2 THEN...
50 IF N=3 THEN...
```

Efectivamente, en la línea 20 se ha realizado una depuración para filtrar los valores de **N** superiores a tres e inferiores a uno, ya que al no cumplir ninguna de las condiciones del bloque 30-50, pasarían a la zona reservada de programa, a partir de la 60.

Sin embargo, ¿qué pasará si el operador introduce un valor decimal, bien por descuido o por poner a prueba nuestra habilidad como programadores? Si este valor supera a tres o resulta inferior a uno, es capturado por nuestra depuración de la línea 20, pero, ¿y de no ser así?

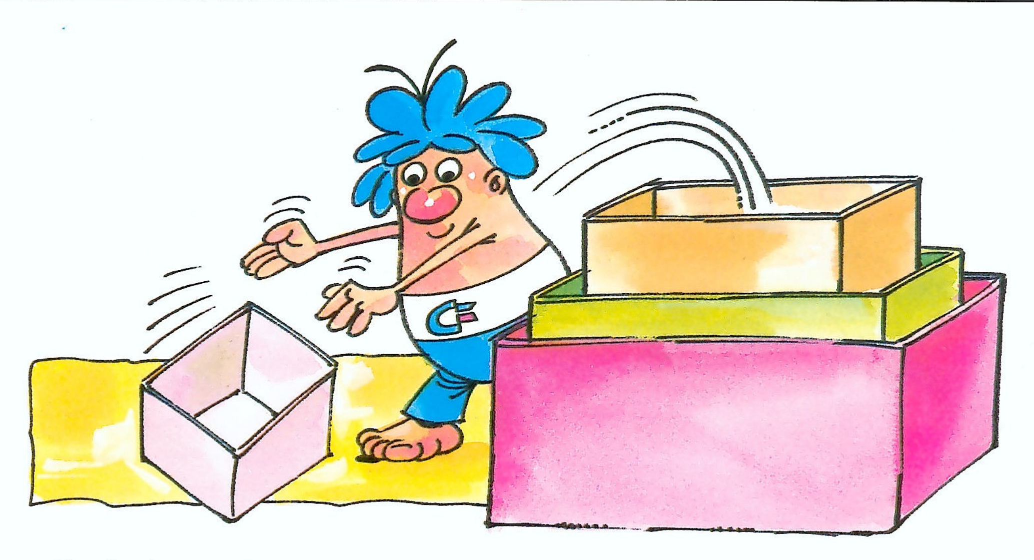
Supongamos que el valor introducido es **2.3.** El valor decimal superara la depuración de la instrucción 20, al estar comprendido entre uno y tres; además, no bifurcará la ejecución del programa hacia ninguno de los números de instrucción propuestos en el bloque 30-50, puesto que el valor introducido no es ni uno, ni dos, ni tres. Por eliminación, el programa continuará su ejecución a partir de la línea 60, con resultados impredecibles.

Con el empleo de la función **INT** podemos asegurarnos que el valor que llega a la línea 20 sea entero, de forma que no eluda la depuración. Vamos a introducir una nueva instrucción en el programa con el número 15. Para ello, podemos seguir dos sistemas: o bien realizar una depuración en la línea de valores enteros, o bien modificar el valor de **N** para convertirlo incondicionalmente en entero. La línea podría quedar como sigue:

```
15 IF N<>INT N THEN 10
15 N=INT N
```



*Cuando deseamos obtener el «redondeo» sumamos 0.5 al argumento de la función **INT**.*



No existe ningún inconveniente para que las funciones BASIC formen parte del argumento de otras funciones.

En el primer caso como depuración de si el valor es entero o no, se emplea la condición «SI N ES DISTINTO DEL VALOR ENTERO de N». Otro sistema puede ser codificar la instrucción como:

```
15 IF N—INT(N)Ø THEN 10
```

El algoritmo se basa en esta ocasión en que la diferencia entre un valor numérico real y su parte entera se hace únicamente igual a cero cuando este es entero; aunque este sistema de codificación resulta más complejo que los anteriores.

En el segundo caso propuesto se modifica mediante asignación directa el valor de **N** de forma que aseguramos la condición de entero del valor que llega a la depuración de márgenes de la línea 20.

En cuanto a la forma de obtener la parte entera hemos de tener presente que la función **INT** «redondea» por defecto, es decir, toma el valor del número entero menor más próximo. Así, por ejemplo, el valor entero que la función **INT** obtiene del argumento **2.3** es dos, mientras que el hallado de **—2.3** es menos tres.

Cuando en el tratamiento numérico para aplicaciones de gestión deseamos obtener, no la parte entera, sino el «redondeo» de un número, la función **INT** no nos resulta útil por si sola. El sistema para aproximar un número real con decimales al entero más próximo es sumar **.5** al argumento de la función. Así por ejemplo, la línea:

```
PRINT INT(N+.5)
```

Imprime el «redondeo» al entero más próximo del valor de **N,** cualquiera que sea éste. De esta forma, los valores decimales iguales o superiores a **.5** implican la aproximación al entero mayor y los restantes al entero inferior.

Por otra parte cabe destacar que las funciones BASIC pueden formar parte del argumento de otras funciones. Así, por ejemplo, podemos obtener el valor absoluto de la parte entera del valor de una expresión **string:**

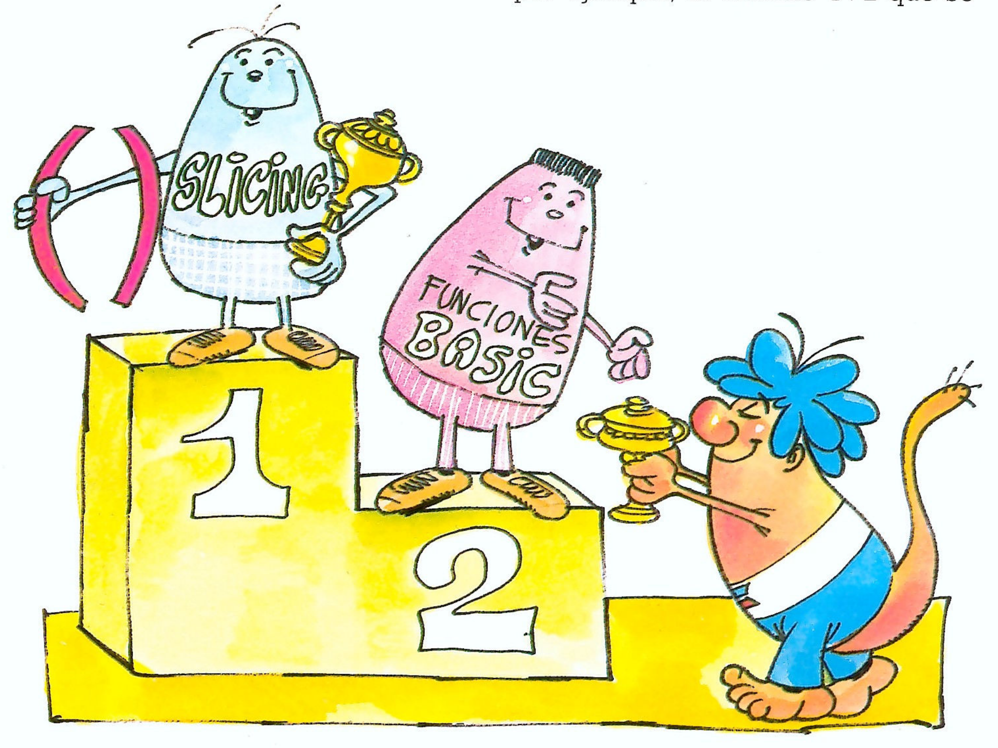
```
PRINT ABS(INT(VAL("—3.5")))
```

Al entrar en este tipo de expresiones de ralativa complejidad pasa nuevamente a primer término el problema de las preferencias, que ya hemos tenido oportunidad de tratar al hablar de los operadores matemáticos. En el caso concreto de las funciones BASIC diremos que gozan de una prioridad muy alta, sólo por debajo de la de los paréntesis y el fraccionamiento de cadenas **slicing.**

Por lo tanto, para escapar a este sistema de prioridades, cuando deseemos que el argumento de la función sea múltiple, deberemos encerrarlo entre paréntesis de la manera que nos convenga. De no proceder de esta forma, ya sabemos que la interpretación se producirá de izquierda a derecha, por ser todas las funciones de idéntico nivel de prioridad.

LA POTENCIACIÓN

La potenciación consiste en multiplicar un número (base) por si mismo tantas veces como indica el superíndice que le sigue (exponente). Así, por ejemplo, el término **4↑2** que se



Las funciones BASIC gozan de una prioridad muy alta, sólo por debajo de la de los paréntesis y el slicing.

lee «cuatro elevado a dos» (o «cuatro al cuadrado»), tiene como resultado **4*=16.**

Lógicamente, debido a la restricción de no poder utilizar la notación habitual de superíndices dentro del texto BASIC, se adopta el criterio de representar esta operación indicando la base seguida del símbolo de la flecha hacia arriba (**↑**) y a continuación su exponente. De esta forma, la operación del ejemplo anterior se escribe en BASIC de la forma: **4↑2.**

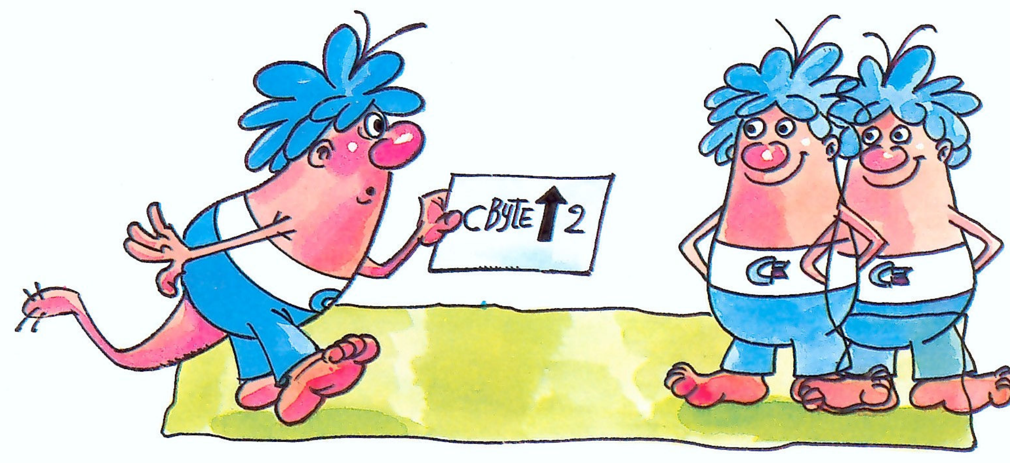
Dentro de las funciones matemáticas es esta de las de uso más frecuente, sobre todo con el exponente dos. La potenciación tiene una operación inversa que nos permite conocer qué base elevada a un determinado exponente proporciona un resultado concreto. Esta operación se denomina radicación, y se representa por una uve en cuya parte superior se expresa el índice de la raíz, unida a una línea de super-rayado debajo del cual se escribe el radicando o argumento de la función ( ).

En el caso concreto del ejemplo anterior, la operación inversa al cuadrado de cuatro es la raíz cuadrada de **16** representada como: **16,** cuyo resultado es cuatro. Así pues, se cumple que dada una base cualquiera **B** y un exponente **E:**

B↑E=X y B= X

Pero al igual que no es posible representar en BASIC la potenciación con su simbología matemática tradicional, menos aún podemos utilizar la notación matemática habitual para la radicación.

Así pues, para el caso concreto de la raíz cuadrada (índice dos), que es sin duda la operación de radicación más común, se utiliza una función BASIC específica: **SQR.** Esta función es la abreviatura de la palabra inglesa **SQuaRe** que significa cuadrado, por tanto, si efectuamos un: **PRINT SQR(16),** obtendremos el resultado de cuatro.



La potenciación consiste en multiplicar un número por sí mismo tantas veces como indica el superíndice que le sigue.

Ahora bien, tanto la potenciación como la radicación se pueden efectuar con cualquier exponente o índice. En el caso de la primera operación expresar un exponente distinto de dos es bien sencillo: no hay más que cambiar el valor escrito tras el símbolo ↑ por el literal o variable numérica que deseemos; sin embargo, en el caso de la radicación las cosas se complican un poco.

Para resolver este problema debemos disponer ante todo de cierta base matemática. Como ya hemos dicho, elevar una base a un exponente equivale a multiplicar dicha base por sí misma tantas veces como indique el exponente; y el BASIC es capaz de tratar exponentes fraccionarios, los cuales resuelven el problema de la obtención de raíces de cualquier índice.



*Para efectuar una raíz cuadrada se emplea la función BASIC específica **SQR.***

Este hecho se fundamenta en una propiedad de las potencias por la cual una base cualquiera **B** elevada a un exponente fraccionario **N/D,** es igual a la raíz de índice **D** de la base **B** elevada a **N,** o lo que es lo mismo:

B↑N/D= B↑N

Por tanto, puesto que cualquier base elevada al exponente unidad tiene como resultado la misma base, podemos particularizar la expresión anterior, asegurando que la raíz n-ésima de cuál número es igual al número elevado al inverso del índice, es decir, a uno partido por éste.

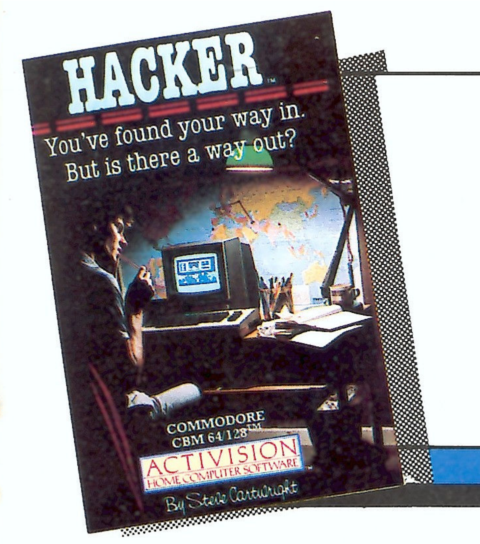
Aplicando esta fórmula, la obtención de la raíz cuadrada puede llevarse a cabo de dos formas diferentes: una por medio de la función específica **SQR** y otra, elevando la base a un medio **(1/2).** Por este sistema se puede obtener cualquier raíz de un número. Por ejemplo, la raíz cúbica de **8** será:

```
PRINT 8↑(1/3)
```

En este caso el exponente de la potencia se encuentra entre paréntesis. De no hacerlo, el BASIC obtendría primero la potencia uno de ocho para después dividir el resultado obtenido por tres, ya que la prioridad de la operación de exponenciación es mayor que la de la división.

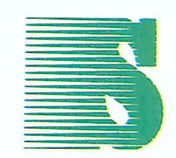
Como última complicación de cálculo, y aplicando estas mismas propiedades, podemos obtener la raíz cúbica de ocho elevado al cuadrado:

```
PRINT 8↑(2/3)
```



Hacker

Esta no es una misión como las demás: algo muy importante (al parecer un valioso documento con información vital) ha sido infiltrado por los agentes enemigos. Tu cometido: conseguir la prueba y llevarla a Washington para asegurar la estabilidad de la compañía «Magma, Ltd.» date prisa, no hay tiempo que perder.



Sobre el contenido del documento no poseemos ninguna información oficial, pero sabemos que se trata de datos secretos acerca de la existencia de una desconocida energía subterránea que podría ser utilizada con fines bélicos. Este documento, elaborado en colaboración con el FBI, es absolutamente secreto (Top Secret, que le llaman los entendidos), y no debe llegar a manos del enemigo. Lo único que tenemos seguro es que el dossier completo no está en posesión de ningún espía, ya que la información que contiene es demasiado importante.

Sabemos, por otra parte, que varios agentes diseminados por todo el mundo, tienen cada uno un fragmento, y que como los métodos de sonsacar una información a tiros han quedado relegados a las películas, no hay más remedio que negociar con ellos y tratar de conseguir el pedazo del que dispongan... por las buenas.

Está claro que una infiltración en un sistema de seguridad se detecta al poco tiempo de producirse, pero la importancia del contenido del documento ha provocado que haya muchos compradores desconocidos que ya están en contacto con los espías. Hay que actuar deprisa, y como las distancias a recorrer son muy grandes, utilizaremos el sistema super-secreto de transporte SRU (unidad subterránea por control remoto). La forma de utilizar estas unidades es, a través de un ordenador conectado con la base central de control de la «Magma, Ltd.», dirigiendo el robot por el laberinto de pasillos subterráneos que recorren todo el globo, y ponernos en contacto con los espías a través de ellos.

Así, sentado ante el teclado de un ordenador, dirigiendo a un robot que, a miles de kilómetros de distancia, se mueve a una velocidad de vértigo entre unas ciudades y otras, el destino de la «Magma, Ltd.» y del planeta, depende únicamente de tí. Eres la única posibilidad de evitar que un valioso documento vaya a parar a manos de quien no debe, bajo ningún concepto, darle ni un solo vistazo. Te necesitamos, no nos falles ahora...

Primera dificultad: introducirse en el control central de la «Magma, Ltd.»

Es obvio que una compañía como ésta posee unos sistemas de seguridad que asombrarían al mismísimo Pentágono, y puesto que todavía no se ha dado la alarma sobre la infiltración, la orden que te permite el acceso al banco de memoria de la «Magma, Ltd.» aún no ha sido procesada, y no podemos perder ni un segundo.

Todo el sistema de protección de la compañía está mecanizado, y ni siquiera una orden expresa por parte del Director que hubiera sido convenientemente verificada podría desactivarlo. Con permiso de éste (más bien, a petición de éste), debes encontrar la forma de entrar en el banco de datos como si de un pirata se tratase. Esta medida es la única salida, porque cuando el ordenador central haya codificado y procesado la orden de franquearte el paso, puede ser ya demasiado tarde.

Esta es, pues, la primera parte de tu misión, que gracias a una relación de amistad que dura ya varios años entre el equipo de redacción de TU MICRO COMMODORE y el diseñador de los sistemas de seguridad de la «Magma, Ltd.» te ofrecemos resuelta.

Durante el transcurso de una invitación a cenar por parte del susodicho genio informático, y tras la decimosegunda cerveza, éste nos confesó que la forma de eludir al ordenador es la siguiente:

En la pantalla aparece el mensaje "Longon please: ", a lo cual debemos responder simplemente pulsando la tecla «H» (+ **RETURN**). El computador volverá a preguntar "Longon please: ", y pulsando unas teclas cualesquiera (+ **RETURN**) el acceso se abrirá.

Nos es asignada una SRU cuyo modelo (AXD-031 0479) debemos apuntar, ya que más tarde nos hará falta recordarlo. Antes de darnos el control del aparato, debemos familiarizarnos con él, moviendo mediante el joystick una cruz que aparece en pantalla, y pulsando disparo sobre el área concreta del robot que se nos pregunte. No debemos cometer ni un solo fallo, ya que la secuencia de preguntas sobre las partes del robot se volverá a repetir hasta que acertemos todas correctamente, y puede llegar a resultar bastante aburrido ver una y otra vez la lista de nombres sobre las piezas del SRU y estarlas disparando. Para facilitar esta cuestión, ofrecemos un utilísimo esquema sobre las partes de la unidad subterránea más importantes (quede claro que aquellos que presumen de



una extrema deportividad en las competiciones de videojuegos, no deben ni mirarlo).

Moviendo el SRU bajo tierra: un laberinto de túneles entre continentes

Cuando hayamos pasado con éxito la prueba de aptitud para el manejo de nuestra unidad de control remoto, un mapa del planeta aparecerá con la posición del SRU señalada en todo momento por una luz parpadeante. Las ciudades en que se ha detectado la presencia de espías implicados en el caso se muestran mediante un recuadro blanco. Debemos hacer llegar hasta ellas el robot a través de una maraña de pasillos de diversas longitudes que se entrecruzan, muchos de cuales no tienen salida.

Obviamente, los diseñadores de juegos tratan de presentar las mayores dificultades a los usuarios, no por fastidiar, sino para que tardemos más en aburrirnos de ellos (cosa que, por otra parte, siempre acaba ocurriendo); esta es la razón de que el trazado de los túneles no aparezca en la pantalla, siendo sólo visibles la dirección de los pasillos que sean accesibles desde nuestra posición. Pero, el equipo de investigación de TU MICRO COMMODORE, al servicio de sus lectores una vez más (aplausos y clamor de agradecimiento) ofrece un mapa del intrincado laberinto de túneles por el cual se desenvuelve el SRU, para hacer más fácil este cometido (mil perdones, señores programadores).

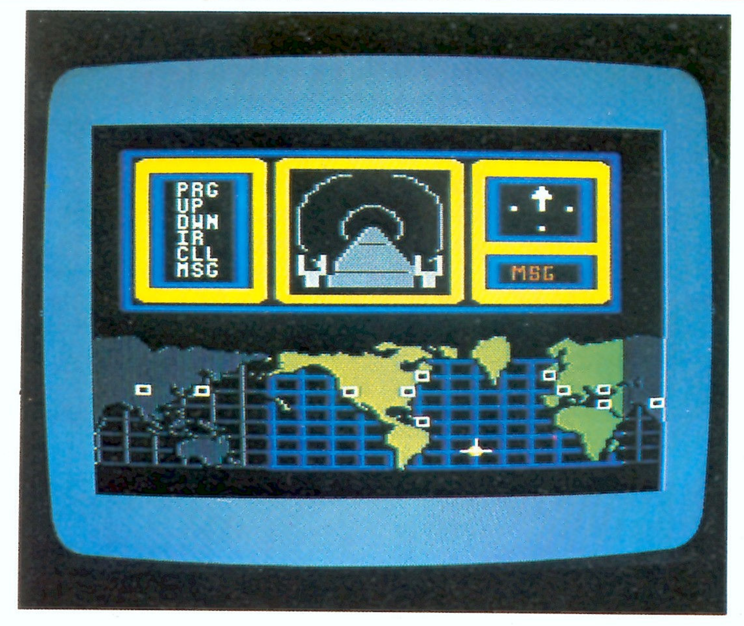
El modo de desenvolverse por los túneles no es muy complicado, pero consigue desesperar progresivamente al jugador que no sabe controlar totalmente la dirección en que realmente desea que se mueva su SRU. En el margen superior derecho de la pantalla hay un recuadro en negro sobre el cual se sitúa una flecha que indica la dirección del movimiento. Si deseamos mover nuestro espía mecánico hacia el Norte (la parte superior del mapa), hay que empujar el joystick en esa dirección. Si sobre la ventanita superior central aparece un muro de rocas, debemos interpretar que en esta parte determinada del trazado no hay ningún tunel que nos lleve hacia el Norte. Habrá, pues, que dar un rodeo, y encontrar un pasillo que discurra en la dirección deseada.

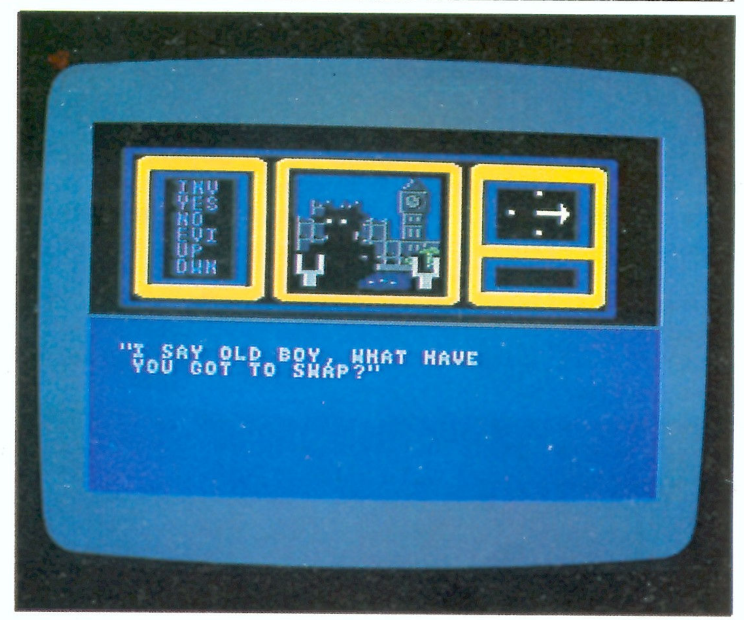
Si en la pantallita superior aparece la vista en perspectiva de un tunel, quiere decir que podemos ir en la dirección deseada, al menos mientras el pasillo no se acabe. Volviendo al ejemplo anterior, si al empujar el joystick hacia adelante vemos un tunel, podemos mantener el mando en esta posición para seguir la línea del tunel hasta que termine, o hasta llegar a una altura que consideremos útil según el lugar al cual nos encaminemos. Este proceso se puede dar a la inversa, es decir, tirando del joystick hacia atrás acortamos la línea hasta el lugar deseado. Una vez hayamos decidido la dirección y longitud del tunel pulsamos disparo, y el SRU avanzará por el tunel hasta el lugar señalado.

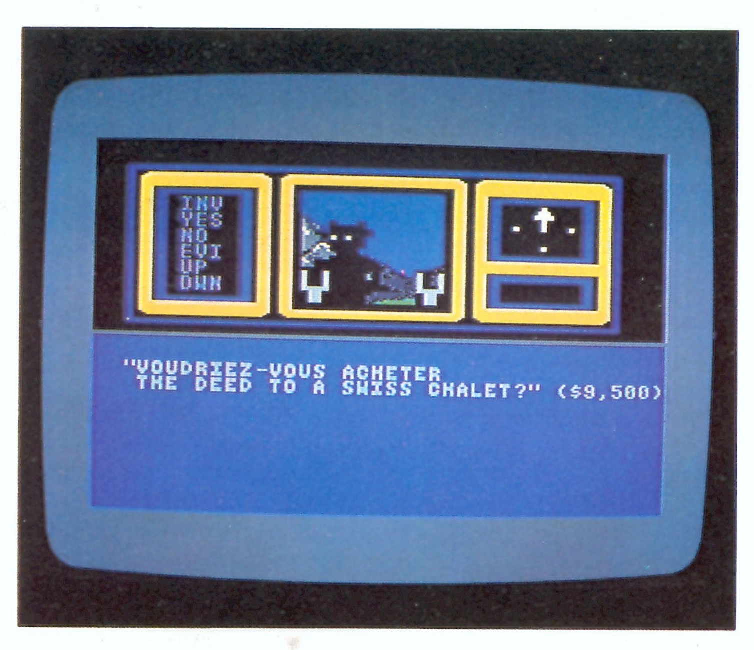
Un hecho que suele confundir mucho al principio es que si inclinamos el mando a la derecha, a veces la vista de las rocas da la sensación de que estamos girando en sentido contrario. Para no hacerse lios, hay que entender que lo que realmente importa es la posición de la flecha y la visión por la ventanita. El sentido en el que gira es sólo una forma de nuestro SRU de llegar a enfocar lo que se ve en la dirección elegida.

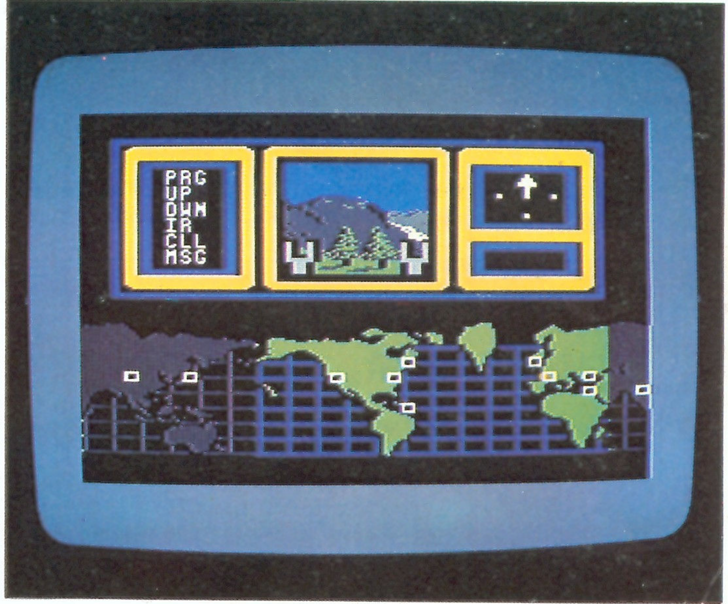
Disponemos además de una lista de comandos para utilizar en nuestro recorrido, y que son de bastante utilidad:

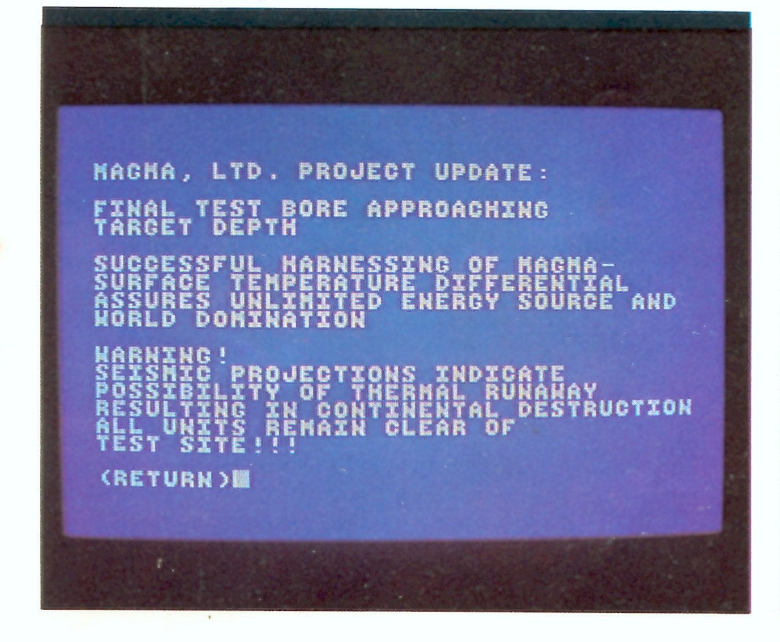
— PRG. Si parpadea indica que el programa que mueve al robot está funcionando.

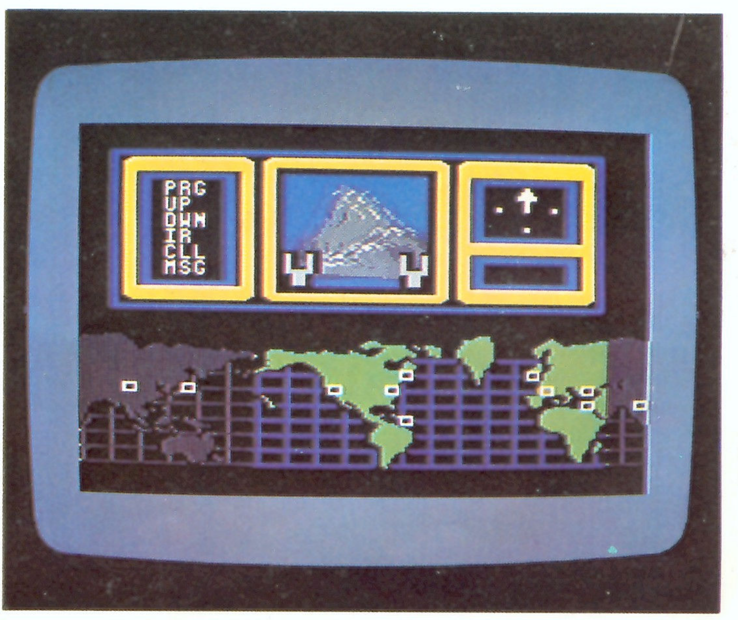

PRG
UP
DWN
IR
CLL
MSG
MSG


INV
YES
NO
EVI
UP
DWN
"I SAY OLD BOY, WHAT HAVE YOU GOT TO SWAP?"


INV
YES
NO
EVI
UP
DWN
"VOUDRIEZ-VOUS ACHETER THE DEED TO A SWISS CHALET?" ($9,500)


PRG
UP
DWN
IR
CLL
MSG


MAGMA, LTD. PROJECT UPDATE:
FINAL TEST BORE APPROACHING TARGET DEPTH
SUCCESSFUL HARNESSING OF MAGMA-SURFACE TEMPERATURE DIFFERENTIAL ASSURES UNLIMITED ENERGY SOURCE AND WORLD DOMINATION
WARNING!
SEISMIC PROJECTIONS INDICATE POSSIBILITY OF THERMAL RUNAWAY RESULTING IN CONTINENTAL DESTRUCTION
ALL UNITS REMAIN CLEAR OF TEST SITE!!!
<RETURN>


PRG
UP
DWN
IR
CLL
MSG

— UP. Al llegar a una ciudad, sirve para subir a la superficie y contactar con el espía.

— DWN. (Down) Lo contrario que en el caso anterior: descendemos de nuevo a los túneles para seguir nuestra búsqueda.

— IR. Debido a la diferencia de horarios entre unos países y otros, puede ocurrir que al llegar a una ciudad sea de noche. Conectando los infrarojos no tendremos problemas de visibilidad.

— CLL. (Call) El espía puede haberse despistado contemplando el paisaje o las nativas de una ciudad, y con este comando le llamamos.

— MSG. Cuando al tiempo que suena un molesto «pi-pi-pi-pi» aparecen estas letras parpadeando, recibimos un mensaje de la base, al cual debemos atender cuanto antes.

Todos estos comandos se manejan con la pulsación de la inicial del nombre (por ejemplo, para subir a la superficie, debemos pulsar «U»), y nuestro SRU hará el resto.

Hay que leer con atención los mensajes que recibamos, echando mano cuando sea necesario del diccionario de inglés que todos tenemos en casa. Algunos de estos mensajes nos avisan del lanzamiento de satélites rastreadores que nos buscan.

Estos aparatos saben que estamos usando los túneles secretos, pero como no tienen la confirmación todavía de que nos está permitido por orden expresa del Director manejar los SRU (¡¡Maldita burocracia!!), se comunicarán con nosotros, en el caso de que seamos detectados, para confirmar nuestra utilización de sus aparatos. Nos harán preguntas acerca de la compañía (recuerda: «Magma, Ltd.», los espacios y la coma cuentan), del modelo que usamos (AXD-0310479, ya dijimos antes que conviene apuntarlo) y una larga serie de cosas, que en el momento que no sepamos contestar, harán que los satélites se enfaden y nos corten la comunicación con e' SRU que utilizamos, y ya nos podemos despedir, a no ser que juguemos otra partida («F1»).

La superficie: contactos con espías y un poco de aire puro para el SRU

Si hemos logrado llegar a una ciudad (es más difícil de lo que parece, así que no te desanimes si no te sale a la primera) y subimos a la superficie, debemos conectar los infrarojos si es de noche y llamar al espía («C»). En la ventana superior por la que antes veíamos si había tunel por el cual avanzar, aparece ahora un bonito paisaje característico de la ciudad donde estemos. Podemos ver entonces al espía, una siniestra figura vestida de negro que nos pregunta qué le ofrecemos a cambio del fragmento que posee. Lo más curioso de esto es el hecho de que cada espía habla en la lengua de su país, y si resulta difícil entender un mensaje en inglés, imagínate un diálogo en griego.

Disponemos de una lista de comandos que, como en el caso anterior, nos ayudan a comunicarnos con el espía a través del SRU. Son los siguientes:

— INV. Ofrecemos los objetos del inventario que llevemos con nosotros, o el dinero del que dispongamos.

— YES. Contestar «SI» a lo que nos pregunte.

— NO. Contestar «NO».

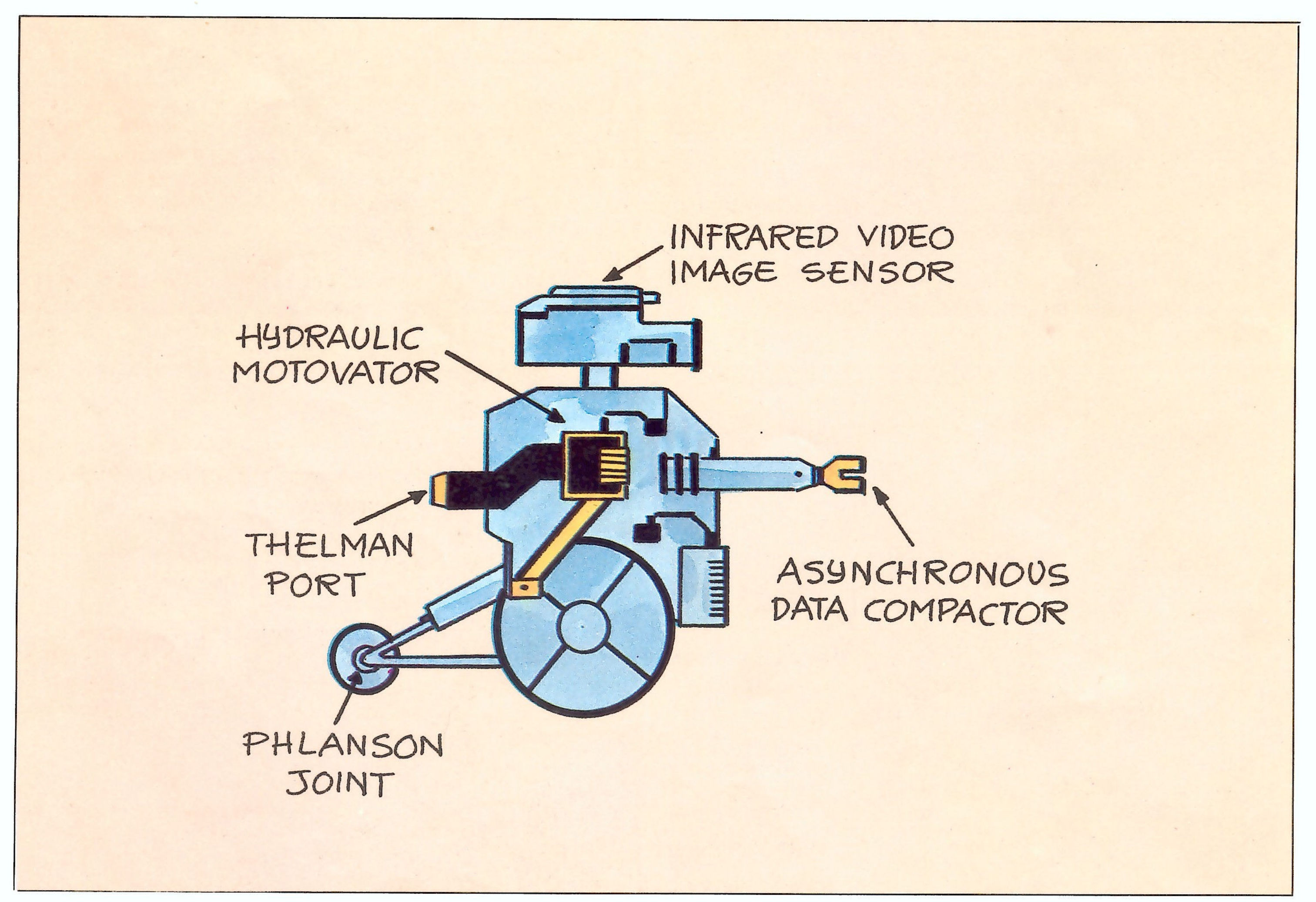
— EVI. (Evidence) Muestra los trozos de documento que tenemos.

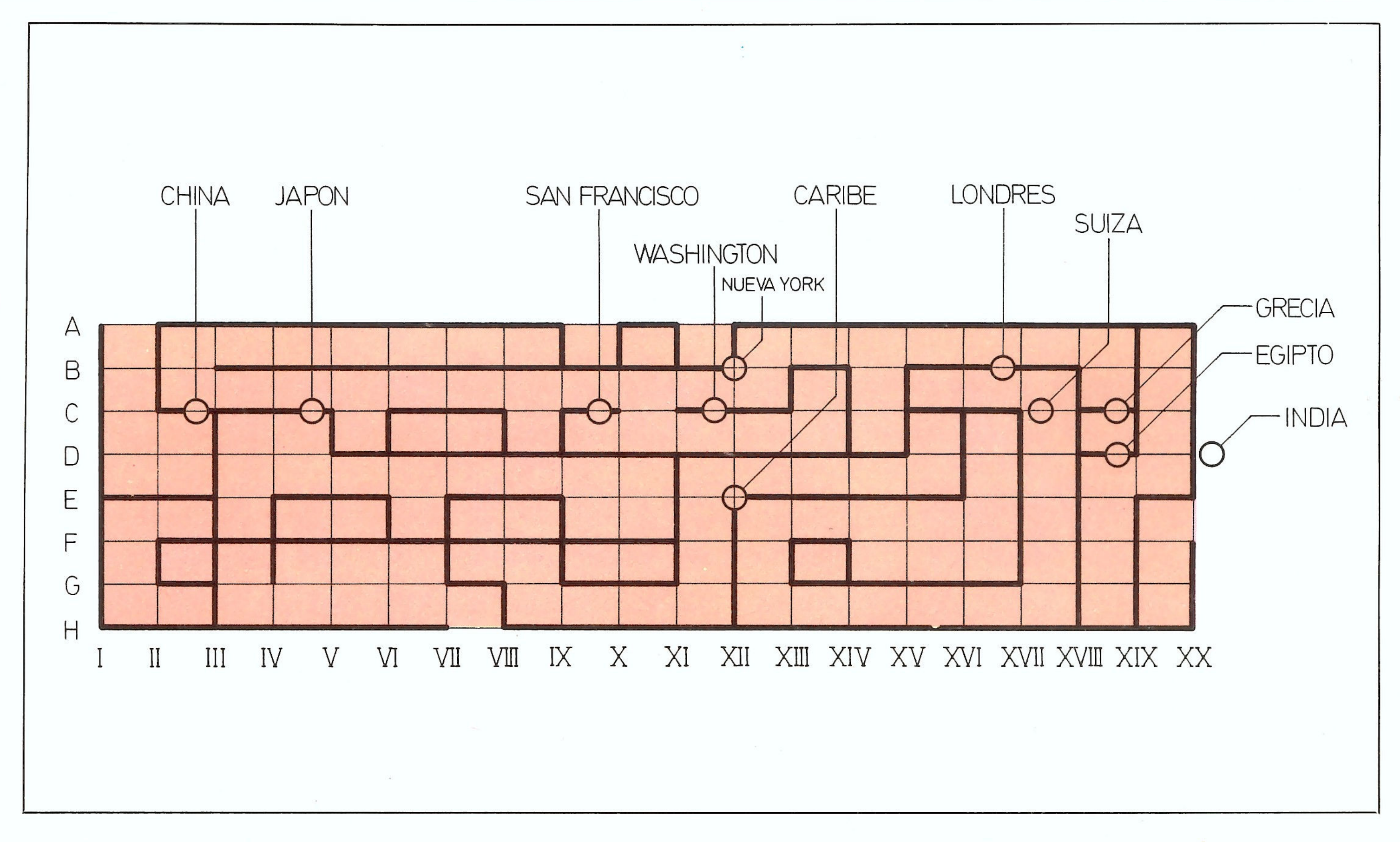
— UP. Lo mismo que en la lista anterior.

— DWN. Idem de idem.

Cuando el espía nos pregunte qué le ofrecemos, pulsamos la tecla «I», para darle dinero. Si no lo quiere, puede seguírsele ofreciendo una lista de cosas que hayamos conseguido por el camino, pulsando esta tecla cada vez para ofrecer una cosa.

Si finalmente acepta nuestra oferta, nos dará su parte del documento, que quedará





guardada en espera de conseguir todas.

Un espía puede también ofrecernos algo para que le compremos (con su sueldo no llegan a fin de mes, su profesión está bastante desconsiderada en nuestros tiempos). Es conveniente acaparar todas las cosas posibles (desde un disco dedicado de los Beatles hasta una simple brújula), porque un espía que no acepte dinero, puede cambiar su fragmento por una joya egipcia, por ejemplo.

A veces, será necesario volver varias veces a una misma ciudad, hasta conseguir tener algo que le guste al caprichoso espía (realmente, algunos lo son, y mucho) para cambiar por el secreto que posee.

La cuestión de los idiomas es fácil de interpretar si consideramos que a todas sus ofertas de venta debemos decir que sí, y que sus respuestas son siempre las mismas para rechazar lo que le proponemos, y en caso de aceptar, coloca un maletín en la pinza del SRU, con lo que no se hace imprescindible entender que está diciendo exactamente.

Un rasgo humorístico es que al llegar a Norteamérica, no nos hablan en inglés, sino en su «peculiar» pronunciación de esta lengua (por ejemplo, no dice «come on», sino «cmon»). Después de deambular por muchos países, supone un alivio acercarse a Sudamérica, porque uno se siente como en casa (España no tiene ninguna ciudad representada, lo que indignará bastante a aquellos que posean un fuerte sentido nacionalista).

El subir a la superficie tiene la ventaja de que en ella, los satélites no nos pueden detectar; así que cuando venga uno hay que ir lo más deprisa posible hacia una ciudad y subir, esperando a que pase de largo.

Para conseguir llevar a cabo nuestra misión es muy conveniente empezar visitando Suiza, y continuar con los demás países.

Hay dos puntos fundamentales que no hay que olvidar nunca:

1. No vendas, bajo ningún concepto, ningún «deed» (escritura) en Egipto.

2. Cuando vayas a China, comprueba que llevas «Ornaments from the ocean» para negociar.

El método a seguir es, por tanto, muy simple: movernos con la mayor soltura posible por los túneles (esto se consigue únicamente a base de mucha práctica), esquivar en la medida de lo posible los satélites, contactar con los espías, y conseguir todo el documento. Una vez lo tengamos, dirigiremos el SRU a Washington, donde los interesados sabrán qué hacer con él (a lo mejor resulta que sólo es una receta para hacer natillas, y con lo que nos ha costado conseguirlo...).

PUNTUACION

Nombre:	Hacker
Precio:	2.300 ptas.
Soporte:	Cinta
Tipo:	Juego de estrategia
Observaciones:	¿Instrucciones?... ... ¿Qué instrucciones?

VEREDICTO FINAL

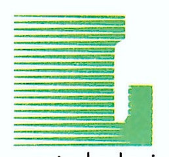
Gráficos:	* * *
Sonido:	* *
Originalidad:	* * * * *
Dificultad:	* * * *
Interés:	* * * * *

La vieja idea de reunir fragmentos, con muchos toques de originalidad

A pesar de lo pobre de las instrucciones, este juego, una vez que se llega a dominar proporciona mucho tiempo de entretenimiento al usuario que lo posee, y por su originalidad en el tratamiento del viejo esquema de los fragmentos dispersos (Impossible Mission, Scarabaeus, y otros muchos) hace que no quepa dentro de esta clasificación. Los gráficos son bastante aceptables, y la presentación en pantalla es magistral. Se sale de los tradicionales esquemas de «tú vas dentro del aparato» o «tú eres el aparato», cambiándolo por «tú controlas el aparato igual que controlas el propio juego».

Códigos mnemónicos del 6502 (1)

Comenzamos hoy el estudio del juego de instrucciones del microprocesador. Las instrucciones disponibles son muchas y muy diversas, y debemos dominarlas todas para poder dominar a la CPU.



La mejor manera de comenzar el estudio del juego de instrucciones es comprendiendo la tabla de la página siguiente. En esta tabla, podemos ver todas las instrucciones disponibles, sus códigos y los modos de direccionamiento de que disponen.

Cada instrucción dispone de algunos direccionamientos, pero no de todos

En la primera columna de la tabla, vemos los mnemónicos de la instrucción, ordenados alfabéticamente, y en la siguiente, una breve descripción de su funcionamiento, que nos servirá como recordatorio de su utilidad.

A continuación, en las siguientes trece (el 6502 no es supersticioso) columnas, una para cada modo de direccionamiento, disponemos de una información muy importante: En primer lugar, bajo la columna COD, tenemos el código de la instrucción escrito en hexadecimal.

Este código representa a la instrucción y el modo de direccionamiento que utiliza; el número de ciclos de reloj necesarios para su ejecución aparecen en la siguiente columna.

Cada instrucción, como ya sabemos, precisa de un operando. El número de bytes que ocupa la instrucción y su operando aparece en la columna adyacente. Observemos que cada instrucción ocupa uno, dos o tres bytes, según el direccionamiento que se les aplique.

Observemos también que algunas instrucciones disponen de un único modo de direccionamiento, mientras otras disponen de varios.

Estas tres columnas se repiten trece veces, una para cada modo de direccionamiento.

Las instrucciones afectan al registro de estado, o a parte de él

En la columna final, podemos ver una representación de los ocho bits del registro de estado del microprocesador. Al hablar de la estructura de la CPU, ya vimos que los ocho bits de este registro se activan independientemente, en respuesta a las operaciones del microprocesador.

En esa columna se indica con un asterisco los bits que pueden ser afectados por la instrucción. Los bits no señalados simplemente no son afectados por esa operación.

Con esta tabla, disponemos de un punto de referencia para el estudio de los mnemónicos del 6502.

Clasificación de las instrucciones según su función

Para estudiar las instrucciones del microprocesador, podemos seguir varios métodos. Podríamos ir describiendo cada instrucción por separado, tal vez en orden alfabético, o también, y esto es lo que haremos, podríamos estudiar grupos de instrucciones que pueden clasificarse bajo una función común.

Podemos distinguir, en principio, cuatro grandes grupos que engloban a todas las funciones disponibles:

1. INSTRUCCIONES DE TRANSFERENCIA DE DATOS: Sirven para leer o escribir entre la memoria y un registro o entre dos registros del microprocesador. No modificamos ni manipulamos el dato, sino que simplemente lo movemos, lo «transferimos» entre dos puntos del ordenador. Incluiremos en este grupo a las instrucciones que se encargan del manejo de la pila o stack, pues realmente, la pila no es más que un medio de almacenar información temporalmente.

2. INSTRUCCIONES DE OPERACION: Realizan operaciones, tanto aritméticas como lógicas, entre una posición de memoria y un registro, o bien, como veremos, operan sobre un registro o una memoria individualmente. Instrucciones de este tipo son la suma, resta, rotación, etc.

3. INSTRUCCIONES DE FLUJO DE

PROGRAMA: Estas instrucciones abarcan toda la gama de saltos y bifurcaciones condicionales o incondicionales en el programa, los equivalentes a los **GOTO, GOSUB, RETURN** e **IF...THEN GOTO** del BASIC.

4. INSTRUCCIONES DE MODIFICACION DEL REGISTRO DE ESTADO: Modifican un bit del registro de estado, permitiendo acceder a los diferentes modos de operar del microprocesador, como modo hexadecimal/decimal, permitir o inhibir interrupciones, etc.

Y comenzamos, como no, por el principio...

Instrucciones de transferencia de datos: Información de aquí para allá

Dentro de las instrucciones de transferencia, podemos distinguir varios tipos que veremos por separado.

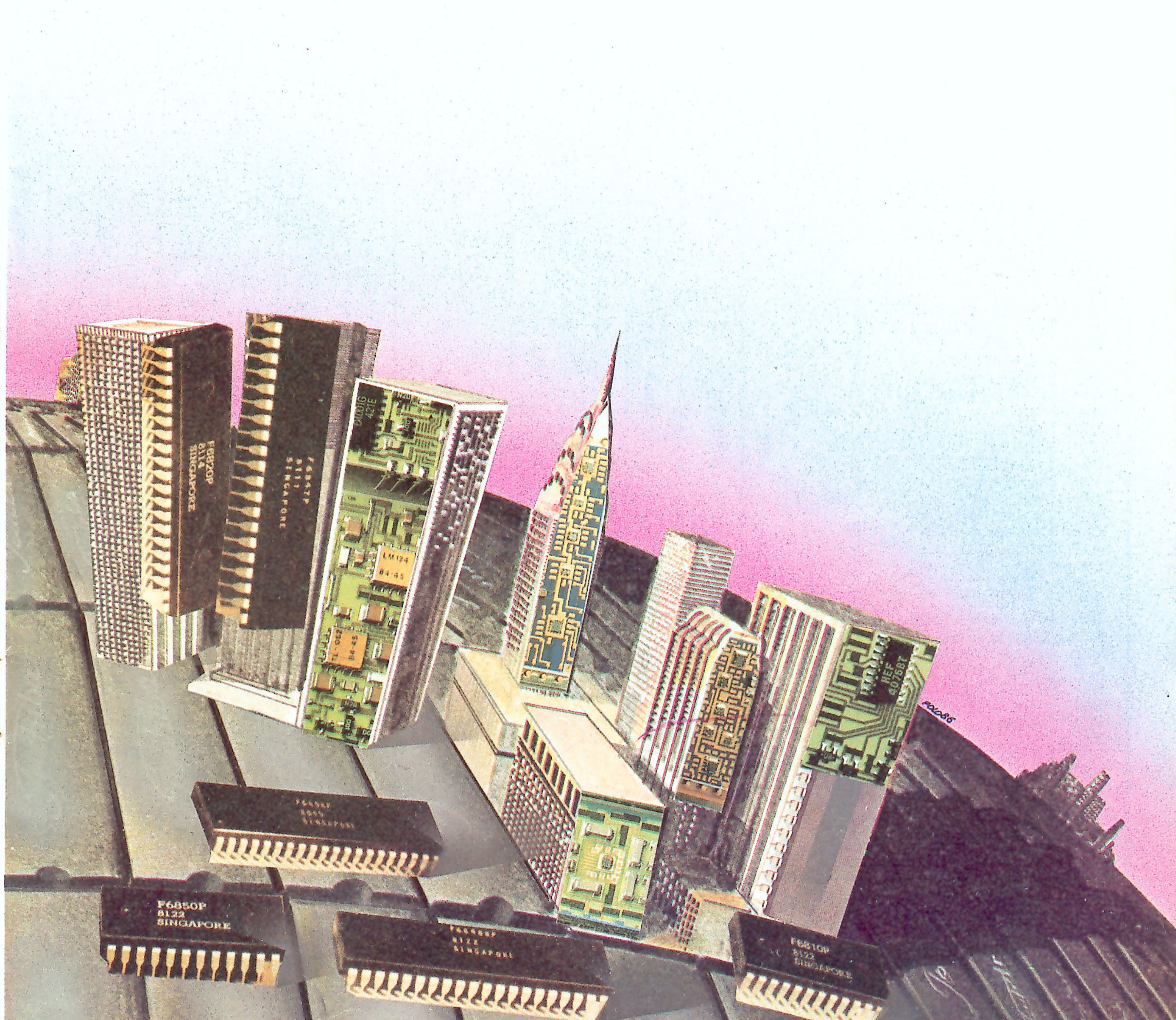
Las instrucciones de transferencia entre la memoria y un registro sirven para mover el contenido de cierta posición de memoria a uno de los registros de la CPU, tenemos la instrucción LoaD (cargar, en inglés): LDA significa cargar de memoria al acumulador, LDX hace lo mismo con el registro X y LDY con el registro Y.

Como ya sabemos, esta instrucción debe seguirse de un operando que indique la posición de memoria que deseamos leer; así, LDA $FF00 significa cargar en el acumulador el contenido de la posición de memoria $FF00. Todas las formas de gestionar la memoria ya fueron vistas al explicar los modos de direccionamiento.

En la tabla de mnemónicos adjunta podemos ver que las instrucciones de este tipo disponen de los modos de direccionamiento inmediato, absoluto, página cero, y los modos indexados indirectos y absolutos. Obviamente, el registro X debe ser indexado por Y, y viceversa. La consulta de la tabla de códigos nos indicará si el modo que deseamos se halla disponible.

Hemos de hacer notar que al cargar el contenido de la memoria en un registro, el contenido anterior de éste desaparece, por lo cual debemos almacenarlo en caso de que sea necesario para un uso posterior, según las necesidades de nuestro programa.

El registro de estado se ve afectado en los bits Z y N según el byte que hayamos cargado (bits de cero y negativo). Esto nos permite, como ya veremos, hacer saltos condicionales tras una instrucción de carga.



No sólo leer, sino también escribir en la memoria

Además de leer la memoria, podemos escribir en ella. Para hacerlo, disponemos de la instrucción STore (en inglés almacenar): STA almacena el contenido del acumulador en la posición de memoria indicada, STX almacena el registro X y STY el registro Y.

El contenido del registro desde el cual escribimos no se modifica; podemos almacenar ese valor y usarlo luego en una operación cualquiera, pues no se pierde, sólo se copia en la memoria y permanece intacto en el registro original.

Los modos de direccionamiento disponibles son los mismos que para las instrucciones de lectura, excepto el modo inmediato. El registro de estado no se modifica por estas instrucciones.

Si intentamos escribir en una posición de memoria ocupada por ROM, la cual como sabemos puede ser leida pero no escrita, no ocurrirá nada anormal, sino que, simplemente, el contenido de la posición ROM permanecerá inalterado. Tranquilos pues, ya que no podemos deteriorar la ROM ni desde có-

M O D O S   D E   D I R E

		INMEDIATO			ABSOLUTO			PAGINA CERO			ACUMULADOR			IMPLICITO			(IND,X)			(IND),Y		
MNEMONICO	OPERACION	COD	n	#	COD	n	#	COD	n	#	COD	n	#	COD	n	#	COD	n	#	COD	n	#
ADC	A+M+C → A (1) (3)	69	2	2	6D	4	3	65	3	2							61	6	2	71	5	2
AND	A and M → A (1)	29	2	2	2D	4	3	25	3	2							21	6	2	31	5	2
ASL	C ← [] ← 0				0E	6	3	06	5	2	0A	2	1									
BCC	SALTA SI C = 0 (2)																					
BCS	SALTA SI C = 1 (2)																					
BEQ	SALTA SI Z = 1 (2)																					
BIT	A and M (2)				2C	4	3	24	3	2												
BMI	SALTA SI N = 1 (2)																					
BNE	SALTA SI Z = 0 (2)																					
BPL	SALTA SI N = 0 (2)																					
BRK	Interrup. por programa													ØØ	7	1						
BVC	SALTA SI V = 0 (2)																					
BVS	SALTA SI V = 1 (2)																					
CLC	HACE C = 0													18	2	1						
CLD	HACE D = 0													D8	2	1						
CLI	HACE I = 0													58	2	1						
CLV	HACE V = 0													B8	2	1						
CMP	Compara con acumul.	C9	2	2	CD	4	3	C5	3	2							C1	6	2	D1	5	2
CPX	Compara con Reg. X	E0	2	2	EC	4	3	E4	3	2												
CPY	Compara con Reg. Y	C0	2	2	CC	4	3	C5	3	2												
DEC	M–1 → M				CE	6	3	C6	5	2												
DEX	X–1 → X													CA	2	1						
DEY	Y–1 → Y													88	2	1						
EOR	A ex – or M → A (1)	49	2	2	4D	4	3	45	3	2							41	6	2	51	5	2
INC	M+1 → M				EE	6	3	E6	5	2												
INX	X+1 → X													E8	2	1						
INY	Y+1 → Y													C8	2	1						
JMP	Salta de programa				4C	3	3															
JSR	Salto a subrutina				20	6	3															
LDA	M → A (1)	A9	2	2	AD	4	3	AS	3	2							A1	6	2	B1	5	2
LDX	M → X (1)	A2	2	2	AE	4	3	A6	3	2												
LDY	M → Y (1)	A0	2	2	AC	4	3	A4	3	2												
LSR	O → [] → C				4E	6	3	46	5	2	4A	2	1									
NOP	NO OPERAR													EA	2	1						
ORA	A or M → A	09	2	2	0D	4	3	05	3	2							Ø1	6	2	11	5	2
PHA	A → MS; S–1 → S													48	3	1						
PHP	P → MS; S–1 → S													Ø8	3	1						
PLA	S+1 → S; MS → A													68	4	1						
PLP	S+1 → S; MS → P													28	4	1						
ROL	← [] ← C				2E	6	3	26	5	2	2A	2	1									
ROR	→ [] → C				6E	6	3	66	5	2	GA	2	1									
RTI	Retorno de interrupción													4Ø	6	1						
RTS	Retorno de subrutina													6Ø	6	1						
SBC	A–M–C̄ → A (1)	E9	2	2	ED	4	3	E5	3	2							E1	6	2	F1	5	2
SEC	HACE C = 1													38	2	1						
SED	HACE D = 1													F8	2	1						
SEI	Hace I = 1													78	2	1						
STA	A → M				8D	4	3	85	3	2							81	6	2	91	6	2
STX	X → N				8E	4	3	86	3	2												
STY	Y → M				8C	4	3	84	3	2												
TAX	A → X													AA	2	1						
TAY	A → Y													A8	2	1						
TSX	S → X													BA	2	1						
TXA	X → A													8A	2	1						
TXS	X → S													9A	2	1						
TYA	Y → A													98	2	1						

– Todos los códigos en Hexadecimal.
(1) Añadir un ciclo si se cambia de página.
(2) Añadir un ciclo si se salta a la misma página.

dos ciclos si se cambia de página.
(3) Si está en modo decimal, Z no se usa.
– M = Memoria

digo máquina (aunque si con un buen martillo, lo cual de todas formas no resulta aconsejable...).

Los mnemónicos reciben este nombre pues sirven para recordar

Un hecho que los más perspicaces habréis observado es que el código mnemónico de la instrucción se forma abreviando la expresión inglesa que la describe: LDA es una abreviatura de LoaD Accumulator (carga acumulador); STX aparece abreviando STore X (almacena X), etc.

De aquí que los códigos se llamen mnemónicos, pues son eso, códigos que ayudan a recordar la utilidad de la instrucción (si eres inglés; a los demás mortales, sólo nos ayuda a aprender ese idioma. A ver si a alguien se le ocurre diseñar un ensamblador en castellano).

En lo sucesivo, siempre que sea posible se indicará la expresión de la cual nazca el mnemónico en cuestión. La próxima semana continuaremos con el estudio de las instrucciones de transferencia de datos.

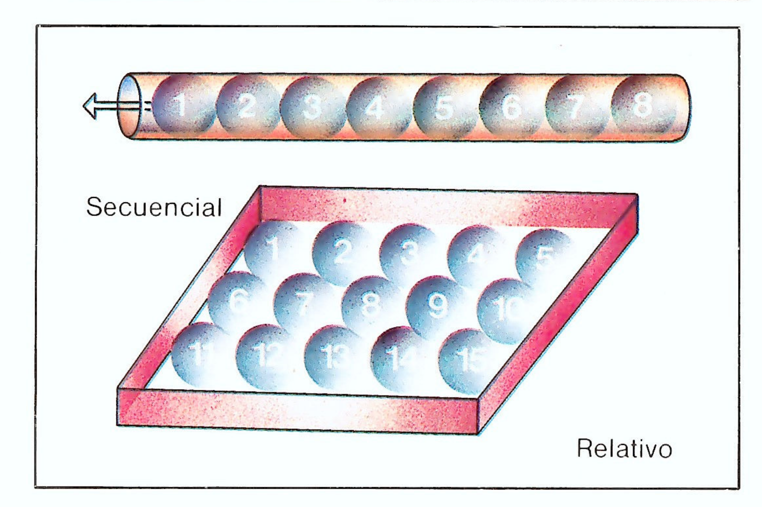
C C I O N A M I E N T O

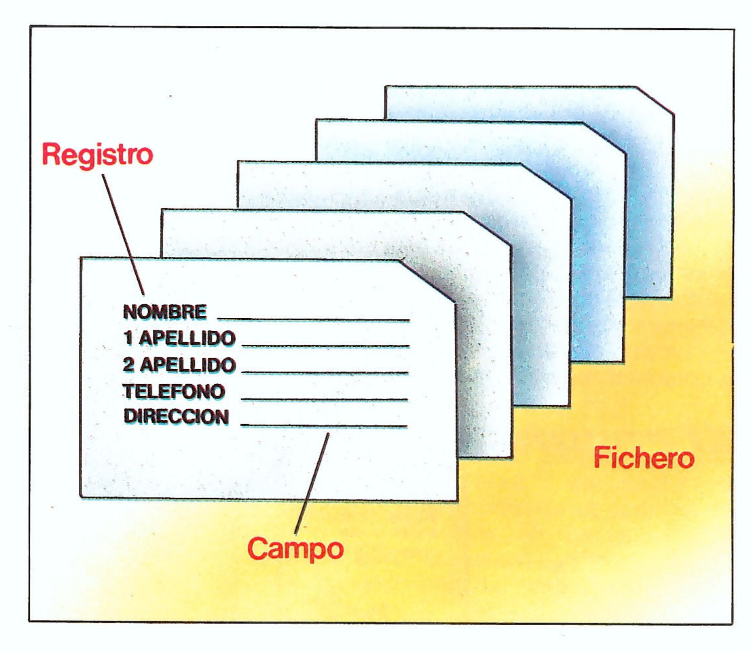
PAG. CERO, X			ABSOLUTO, X			ABSOLUTO, Y			RELATIVO			INDIRECTO			PAG. CERO, Y			EFECTO SOBRE REGISTRO DE ESTADO							
COD	n	#	COD	n	#	COD	n	#	COD	n	#	COD	n	#	COD	n	#	N	V	B	D	I	Z	C	
75	4	2	7D	4	3	79	4	3										*	*				*	*	ADC
35	4	2	3D	4	3	39	4	3										*					*		AND
16	6	2	IE	7	3													*					*	*	ASL
									9Ø	2	2														BCC
									BØ	2	2														BCS
									FØ	2	2														BEQ
																		M_7	M_6				*		BIT
									3Ø	2	2														BMI
									DØ	2	2														BNE
									1Ø	2	2														BPL
																				*		*			BRK
									5Ø	2	2														BVC
									7Ø	2	2														BVS
																								*	CLC
																								*	CLD
																						*			CLI
																			*						CLV
D5	4	2	DD	4	3	D9	4	3										*					*	*	CMP
																		*					*	*	CPX
																		*					*	*	CPY
D6	6	2	DE	7	3													*					*		DEC
																		*					*		DEX
																		*					*		DEY
55	4	2	5D	4	3	59	4	3										*					*		COR
F6	6	2	FE	7	3													*					*		INC
																		*					*		INX
																		*					*		INY
																									JMP
												6C	5	3											JSR
B5	4	2	BØ	4	3	B9	4	3							B6	4	2	*					*		LDA
						BE	4	3										*					*		LDX
B4	4	2	BC	4	3													*					*		LDY
56	6	2	5E	7	3													*					*	*	LSR
																									NOP
15	4	2	ID	4	3	19	4	3										*					*		ORA
																									PHA
																									PHP
																		*					*		PLA
																		RECUPERADO DE LA PILA							PLP
36	5	2	3E	7	3													*					*	*	ROL
76	6	2	7E	7	3													*					*	*	ROR
																		RECUPERADOS DE LA PILA							RTI
																									RTS
FS	4	2	FD	4	3	F9	4	3										*	*				*	*	SBC
																								*	SEC
																					*				SED
																						*			SEI
95	4	2	9D	5	3	99	5	3																	STA
															96	4	2								STX
94	4	2																							STY
																		*					*		TAX
																		*					*		TAY
																		*					*		TSX
																		*					*		TXA
																									TXS
																		*					*		TYA

- **A = Acumulador**
- **MS = Memoria de Stack (Pila)**
- **n = Número de ciclos de reloj**
- **# = Número de Bytes**
- **M7 : BIT 7 de memoria**
- **M6 : BIT 6 de memoria**

¿Relativos? ¡Sí, gracias!

Prácticamente todo el mundo sabe lo que es un fichero secuencial, o al menos, ha oído hablar de ellos. Los que quizá no sean tan conocidos, son los ficheros relativos.






a difusión tan extendida de los ficheros secuenciales, se debe a dos grandes razones: son fáciles de usar y se pueden usar tanto en disco como en cinta. Los ficheros relativos, son un poco más complicados, pero tienen ventajas importantes sobre los secuenciales: no se necesita cargar el fichero completo; no utilizan mucha cantidad de memoria del ordenador y se tiene acceso directo a la zona de información que nos interesa (registro).

A pesar de estas ventajas su uso está poco extendido, si bien es cierto que el BASIC 2.0 de Commodore no se presta demasiado para tales tareas, pero tampoco nos podemos quejar, ya que por lo menos, no es misión imposible.

Las limitaciones que encontramos en su uso, son:

– Necesidad de una unidad de disco.
– Los registros deben ser de longitud fija.
– Puede ser necesario llevar un fichero secuencial de índice, paralelo al fichero relativo.

Pese a todo, las ventajas superan con mucho a los inconvenientes, así pues vamos a ver el ABC de los ficheros relativos.

Lo primero que necesitamos saber es la longitud de los registros a emplear, y cuántos vamos a utilizar. La longitud puede ser desde 1 hasta 254 bytes por registro. Para poder leer esta información con **INPUT#**, se necesitará que no supere los 88 carácteres (bytes), y que se hayan escrito con un RETURN al final (**PRINT#** sin coma o punto y coma al final); para leer los de mayor longitud se necesitará un **GET#**. Supongamos que vamos a crear un fichero de 80 carácteres por registro y de 5 campos por registro, usaremos:

```
OPEN N,D,C,"NOMBRE,L,"+CHR$(LR+5)
```

Donde...
N: Número de fichero.
D: Dispositivo.
C: Canal (2-24).
L: Indica que a continuación viene la longitud del registro.
LR: Longitud del registro, en nuestro caso es 85 debido a la suma del número de campos por registro.

Pueden estar abiertos a la vez un fichero relativo y otro secuencial; pero nunca dos o más ficheros relativos.

Para preparar el fichero, habrá que grabar algo (CHR$(255)) en el último registro a utilizar, para lo cual, enviaremos un comando de posicionamiento a través del canal de órdenes del «floppy»; veamos un ejemplo:

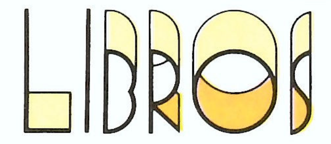
```
OPEN 3,8,15:PRINT#3,"P"+CHR$(N+96)+
    CHR$(BB)+CHR$(BA)+CHR$(BY)
PRINT#N,CHR$(255)
```

Con estas órdenes hemos preparado un fichero de **BA+BB*256** registros, a través del fichero número 3, de una longitud de 80 carácteres por registro; grabando un **CHR$(255)** en el primer byte (**BY=1**) del último registro. Por supuesto, BA y BB son el byte alto y byte bajo del número total de registros (NR):

```
BA=INT(NR/256): BB=NR–BA*256
```

Es más fácil de lo que parece. Inténtalo y lo verás.

Para hacer un registro de varios campos, no debemos olvidarnos de rellenar con blancos el espacio entre campos. Por ejemplo: queremos guardar en los registros anteriores fichas que constan de: Nombre (20 c), Apellido 1 (20 c), Apellido 2 (20 c), Localidad (10 c), Teléfono (10 c); así para: Pedro Uralde Gómez Madrid 7112l 3481, habrá que guardar lo siguiente: Pedro + 15 blancos, Uralde + 14 blancos, Gómez + 15 blancos, Madrid + 4 blancos y 711213481 + 1 blanco. Con ello, siempre tendremos el comienzo de cada campo en las mismas posiciones relativas dentro de un registro.



Colección Data Becker

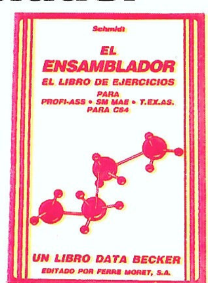
De todos es sabido que los manuales del Commodore 64 dejan mucho que desear, aprovechando este hueco informativo la editorial FERRE MORET ha publicado una extensa colección de libros DATA BECKER traducidos de su homóloga alemana.

Cuando hace pocos años compramos nuestro flamante C-64, a todos se nos hacía la boca agua al pensar lo que podríamos hacer con él, ilusión que fue poco a poco decayendo al comprobar tristemente que éramos incapaces de realizar absolutamente nada de lo que se nos prometía; ¿dónde estaba aquel fabuloso sonido?, ¿y aquellos gráficos multicolor de alta resolución?, ¿dónde se escondían los famosos duendecillos (sprites)?, pues bien, aparte de todo eso nuestro C-64 puede hacer más todavía, para saber cómo nada mejor que consultar con libros especializados como los que describimos aquí.

Debemos pensar que esta colección tiene un sentido fundamentalmente práctico, por lo que el tipo de letra no es precisamente el mejor (parece confeccionado con una impresora tipo margarita y un procesador de texto), sin embargo, el objetivo lo cumple a la perfección; podremos estrujar literalmente a nuestro C-64 hasta sacarle el máximo partido posible, con los ejemplos que vienen nos daremos cuenta de lo que nuestro ordenador es capaz de hacer.

La colección consta de varios volúmenes, no sólo dedicados al C-64, sino también a sistemas MSX, Amstrad y Spectrum; vamos a comentar algunos títulos dedicados al C-64:

El ensamblador



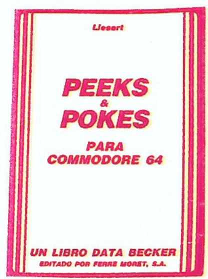
Con este volumen, y dominando todo lo que se nos ofrece en los dos anteriores, se nos proporciona la herramienta necesaria para el dominio completo del ordenador: los ensambladores, con ellos resultará muchísimo más fácil la programación y depuración de todo lo confeccionado en este difícil lenguaje de programación. Se nos brindan ensambladores tales como el PROFI. ASS, SM MAE y .EX.ASS.

El manual del cassette



Todo lo que hay que saber sobre el periférico de almacenamiento más difundido: el Datassette de Commodore, parece increíble que con el datassette se pueda llegar a hacer tanto de forma tan sencilla, como controlar el motor desde programa con programas de demostración como Autostart y Catálogo que se encarga de buscar y cargar el programa automáticamente; también se nos ofrece un «turbo» para acelerar los procesos de carga, cómo ajustar los cabezales, instalación de un altavoz de control, copias a disco, etc.

PEEKS y POKES



Manejando adecuadamente estos comandos podremos hacer desde el BASIC cosas insospechadas de hecho es la única manera de hacer frente a las mejores características del 64 desde BASIC; por ello se indica la estructura del 64 a través de su sistema operativo, intérprete BASIC, página cero, punteros importantes y stacks, caracteres y sprites, programación de interfaces, desactivación de las interrupciones, etc.

64 consejos y trucos



Este libro no es en realidad un libro, son dos, toda una gozada de trucos, recetas y consejos con más de 650 páginas dedicadas a ello. Entre muchísimos más cabe destacar temas como: La simulación del ratón con joystick, conexión de una impresora Centronics a través del Port de Usuario, cómo realizar multitarea en el C-64, el CP/M en el 64, partición de la pantalla, cómo hacer los famosos «scrolls», modificación del sistema operativo a voluntad, modificación del teclado por soft, volcados de pantalla, trabajar con dos pantallas independientes, hacer más confortables las gráficas, menajar todo tipo de interrupciones, usar rutinas del sistema operativo, sonido y por supuesto muchísimos pequeños trucos de uno o varios **PO-**

KEs o **SYS**s, tan populares por sus efectos rápidos y espectaculares. En definitiva, son dos libros que no deben faltar entre los de tu biblioteca informática, si por lo menos eres uno de esos que se enorgullecen de tener un 64.

Lenguaje máquina para Commodore 64



Contiene todo lo necesario para comenzar nuestra andadura por ese mundo arduo y fascinante a la vez del código máquina, con la introducción a todas las instrucciones disponibles por el 6510 (en realidad las únicas que el microprocesador entiende), con ejemplos, programas en código máquina y el manejo del ensamblador. También contiene un simulador de código máquina paso a paso.

Lenguaje máquina para avanzados

Esto sí que es fuerte. Cuando ya dominemos por completo el código máquina (libro anterior), podremos introducirnos en el aprendizaje profesional de nuestro ordenador: el Commodore 64. Este libro nos guía por los intrincados caminos de la programación de bajo nivel, aprovechando al máximo todo lo que el 64 es capaz de ofrecernos; veremos el complejo sistema numérico interno, el fundamento de las interrupciones y de los timers y nos introduciremos con el sistema operativo del 64, vectores, ampliación de comandos, etc.

Todo sobre el Floppy 1541



Este libro resulta fundamental para los usuarios de una unidad de disco. Con él aprenderemos no sólo a usar la unidad correctamente, sino incluso a programarla; ya sabemos que la 1541 dispone de su propio microprocesador 6502, así como de memoria RAM y ROM. Veremos cómo usar perfectamente los ficheros, los mensajes de error, directorio, etc., con un montón de programas y rutinas auxiliares.

Robótica para su Commodore 64

Construye tú mismo un robot sin grandes gastos, introdúcete en ese misterioso mundo de la cibernética, con todo lo necesario para el desarrollo de la tarea: como programar el Port del Usuario, sensor de infrarrojos, alimentación, brazo prensor y otros programas de fácil entendimiento.



Aparte de esto, existen algunos otros títulos sobre temas tan interesantes como: inteligencia artificial, manual escolar para su Commodore 64, programando como es debido y Commodore 128, que es quizás el único libro en castellano que aclara algo para los desorientados propietarios de este modelo.

Mantenimiento y reparación del Floppy 1541



Si eres un poco «manitas», tienes una unidad de disco y quieres ahorrarte disgustos, tiempo y dinero (desgraciadamente en España se gasta mucho en estas tres cosas), este es tu libro. Aprende a alinear la cabeza del disco tú mismo, este problema es la avería más frecuente de los usuarios de la 1541.



CHISPA:

Detener o evitar los listados ha sido siempre la preocupación mayor de los programadores, para impedir que ojos ajenos miren el fruto de nuestra materia gris plasmado sobre el kinescopio (pantalla) o sobre la superficie celulósica tratada o blanqueada (papel). Retórico, ¿no? Pues bien, para evitar todos estos efectos, se han publicado muchos pequeños e inocentes trucos, pero la unión hace la fuerza y aquí va otro más:

Coloca en una línea **REM** el carácter obtenido con SHIFT+L, a partir de ahí se acaba el listado con un ¿SINTAX ERROR.

CHISPA:

Si cargas el registro X con un número del uno al treinta **(POKE 781,N),** y efectúas un **SYS** a la posición 42042, obtendrás un mensaje de error de todos los disponibles en tu C-64. Lo mejor es que pruebes tú mismo todos los posibles.

¿Que para qué sirve esto?, pues nunca se sabe, pero te aseguramos que en nuestra vida de programación, hemos necesitado cosas de lo más dispar.

CHISPA:

Muchos programadores (nosotros entre ellos), cambiamos los colores iniciales del ordenador al encender por una combinación mejor (para el gusto de cada uno). En caso de que desees todo lo contrario, es decir, volver a la pantalla azul oscuro con carácteres azul claro, borrando la pantalla de paso, efectúa **SYS58648.**

CHISPA:

Si quieres que en un momento dado el ordenador emita el mensaje SEARCHING FOR Nombre, deberás colocar los carácteres ASCII del nombre en una posición de memoria y efectuar los siguientes **POKEs:**

POKE 183, Longitud del nombre
POKE 187, Byte bajo de la posición del nombre
POKE 188, Byte alto de la posición del nombre
SYS 62899

CHISPA:

Quizá alguna vez te hayas encontrado con la necesidad de colocar una cierta cadena (string) a partir de una posición de memoria, yo lo hago así en modo directo:

1. A$="TU PROPIO NOMBRE"
2. FORI=1TOLEN(A$):POKE49151+I,ASC(A$,I,1): NEXT

Esto coloca la cadena "TU PROPIO NOMBRE" en forma de código ASCII a partir de la posición de memoria 49152.

CHISPA:

Los mensajes de comunicación entre el ordenador y el usuario, al manejar los periféricos de almacenamiento disco o casete, pueden estar a tu disposición con la siguiente línea:

POKE782,N:SYS61743

Donde N puede tomar los siguientes valores: 0:"I/O ERROR = ", 12:"SEARCHING", 23:"FOR", 27:"PRESS PLAY ON TAPE", 46:"PRESS RE-CORD & PLAY ON TAPE", 73: "LOADING", 81:"SAVING", 89: "VERIFYING", 99:"FOUND" y 106:"OK".

CHISPA:

Hay veces en las que las tenemos una cadena en la memoria del ordenador y queremos visualizarla, si nos hemos tomado la molestia de terminar la cadena con un carácter 0, la podremos extraer fácilmente con:

POKE780,BB:POKE782, BA:SYS43806

BB es el byte bajo y BA es el byte alto, prueba por ejemplo con los valores bb=96 y ba=228.

CHISPA:

El procedimiento que vamos a explicar a continuación, sirve para cambiar la dirección de carga de un programa. Si el programa está en código máquina podemos conseguir reubicarlo en otra zona de memoria, si el programa es reubicable deberá funcionar sin problemas; si el programa es en BASIC podremos cargarlo en otro lugar y luego ajustar punteros. El proceso es como sigue:

1. Ejecutar SYS 63276 y cargar la cabecera.
2. POKE829,bb nuevo inicio.
3. POKE830,ba nuevo inicio.
4. POKE831,bb nuevo final + 2
5. POKE832,ba nuevo final.
6. SYS 62849 cargará el resto.

CHISPA:

¿Tienes la extraña necesidad (para el que lo necesite no le será tanto), de efectuar **RUN** a un programa BASIC desde otro en código máquina? Deberás efectuar un par de saltos a rutina:

JRS 42585 (A659 HEX) y
JRS 42926 (A600 HEX).

CHISPA:

Como ya hemos comentado en chispas anteriores, la representación de un entero (0—65535) en la memoria se realiza por medio de dos bytes denominados alto y bajo, para la reconversión a entero usaremos la siguiente línea:

?ba*256+bb

Sin embargo, hay una rutina de la ROM que se encarga de realizar la conversión, para ello habrá que hacer un **POKE** en la posición 780 con el byte alto, otro en la posición 781 con el byte bajo y ejecutar **SYS48589.**

CHISPA:

ay veces en las que las combinaciones de software y hardware producen los efectos más insospechados, uno de ellos se puede apreciar con la pulsación de las siguientes teclas en el C-64: tecla Commodore + tecla de ; + tecla =, pruébalo y verás lo que sucede: el cursor se pone de color azul claro y desaparece si tenemos los colores originales. Curioso, ¿no?

CHISPA:

Continuamos con las rutinas de la ROM para extraer nuestras chispas, cuando cargamos un programa desde cinta y aparece el mensaje FOUND "nombre", podemos continuar con la carga del programa pulsando la tecla Commodore, luego... hay una rutina que realiza una espera hasta la pulsación de esta tecla; muy bien, es esta: **SYS 58592** esperará la pulsación de dicha tecla. Esto es mucho más fácil que escribir las líneas Basic equivalentes.





Desde Barcelona, Miguel Prats, tras confesarse empedernido tecleador de programas, nos formula las siguientes preguntas acerca de nuestro sistema de introducción de programas FUERA ERRORES:
1. ¿No es un poco complicado este sistema?
2. ¿Todos los programas los haréis con dicho sistema?
3. ¿No publicaréis programas en el BASIC normal de COMMODORE?

En el mismo estilo esquemático, muy de agradecer, en la carta de Miguel, vamos a proceder a la contestación:

1. Verdaderamente, teclearse los programas de las revistas, nos parece una aptitud muy loable. Siempre se aprende más «tecleando» que «cargando». Pero es muy triste gastar horas en ello, para que luego el programa no funcione por culpa de algún error. Pues bien, esto es precisamente lo que tratamos de evitar con este sistema, que además, facilita la introducción de líneas conflictivas (en las cuales, los listados de Commodore son especialistas), descifrando el significado de los «reversos», caracteres de control, gráficos y contando caracteres para nosotros.

2. Si, todos los programas para el C-64 incorporan dicho sistema, porque las ventajas superan con creces a los inconvenientes.

3. El BASIC de Commodore sigue siendo el mismo con el sistema «FUERA ERRORES»; lo que ocurre, es que los listados con el «checksum» al final (ese célebre numerito que caracteriza cada línea), pueden parecer un poco extraños. Pero todo ello es en pro de la comodidad de nuestros lectores. Todo es cuestión de acostumbrarse al sistema. ¡Animo! y esperamos que nos sigas «tecleando».

En esta ocasión, la comunicación nos llega desde Córdoba, donde reside Luis Miguel López Doncel, que nos remite la siguiente carta:
«Soy un joven de Córdoba que hace pocas semanas compré un Commodore-16, pero me han surgido grandes dudas que estoy seguro que tú me contestarás, y son las siguientes:
1. ¿Al C-16 se le puede colocar cualquier cassette o necesita uno especial?
2. ¿Puede ser ampliado de 16 a 64? Si la respuesta es afirmativa, ¿cómo?
3. ¿Los juegos en los que pone C-64 son sólo aplicables a este Commodore o también al C16?
Aparte de esto me gustaría que hablaséis no sólo del Commodore 64 sino también del 16 que estoy seguro que también hay muchos usuarios de este segundo.
Me alegro que haya surgido esta gran revista y espero que dure mucho.
Sin más me despido, esperando tu respuesta.

Aclaremos de nuevo punto por punto los problemas que le surgen a Luis Miguel, en su entrada en el mundo de los microinformáticos.

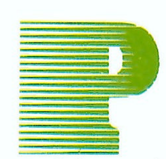
1. No, desgraciadamente no; es algo inconcebible pero así es, en una época en la que todo debería tender a la compatibilidad, el C-16 necesita una grabadora especial; aunque ya las grabadoras de los anteriores Commodore tenían que se especiales, en ese caso no sirve ni la del C-64 (la misma que la del VIC-20 o C-128). Tampoco sirven los joysticks, que deberán ser especiales, y eso que con los joyscticks, la compatibilidad de conectores en casi absoluta.
2. Transcribiendo literalmente las palabras del técnico de Microelectrónica y Control con el cual nos hemos puesto en contacto para resolver tu duda: «No. No hay manera posible».
3. Los juegos son quizás, el software más específicos de cada ordenador, y en este caso no hay excepción. No, no puedes usar juegos diseñados para el 64 en tu C-16. Compra sólo los que indiquen claramente que son para este ordenador.

Jordi Neguera de 13 años, desde Barcelona, nos pregunta cómo hacer un círculo o un cuadrado en alta resolución, así como una explicación del sistema. También pide una sección de trucos e ideas, como por ejemplo: hacer una línea, efectos especiales, hacer moverse un cuadrado por la pantalla, etc.

La verdad, es que todo lo que nos expones, ya existe en TU MICRO COMMODORE, menos lo de los efectos especiales, de lo cual prometemos ocuparnos más adelante. La sección que nos pides se llama CHISPAS, y aparece generalmente en números alternos (¡no hay espacio para todo!). En cuanto a lo demás lo podrás encontrar en diversas secciones de programación aparecidas o por aparecer: rectas y círculos en alta resolución, moviendo sprites por interrupciones (como respuesta a lo del cuadrado móvil).

Procesador de textos EASY SCRIPT

EASY SCRIPT es uno de los procesadores de textos más potentes de los diseñados para la gama de microordenadores COMMODORE, más concretamente para los modelos C-64 y C-128 (en emulación del C-64), con versión comercial residente en cartucho, distribuida por **Casa de Software** de Barcelona y creada por **Precisión Software Ltd.**



Puede decirse de él, en una primera apreciación, que además de las cualidades propias de cualquier procesador, permite el encadenamiento de ficheros de datos, y la técnica de «mail merge». Por ello, puede asegurarse que su capacidad no se encuentra limitada por la cantidad de memoria RAM disponible, a la vez que se convierte en una importante herramienta de gestión a la hora de la confección de cartas personalizadas.

Puesta en marcha del programa

Dado que el programa se suministra en cartucho, no es necesario para ponerlo en funcionamiento más que insertarlo correctamente en la ranura situada en el extremo izquierdo del panel trasero del ordenador, y accionar el interruptor de encendido. Es importante recordar que la conexión o desconexión de los cartuchos debe efectuarse siempre con el ordenador apagado, pues de lo contrario puede dañarse alguno de los componentes del **hardware** de los mismos.

La pantalla de presentación del programa interroga sobre el número de columnas en que se desea trabajar durante la edición del texto, a continuación sobre el soporte magnético que se utilizará para la lectura y almacenamiento de los datos y, por último sobre el tipo de impresora.

Los valores sugeridos por defecto como respuesta a estas opciones son: una anchura de cuarenta columnas, la unidad de disco 1541 y la impresora CBM; valores que serán aceptados automáticamente por la sola pulsación de la tecla **RETURN.** No obstante, pueden definirse otros más convenientes tecleándolos directamente.

Aunque la anchura de edición más comunmente seleccionada es la de cuarenta (pantalla física del C-64), el programa permite trabajar con valores más altos; efectuándose en este caso un desplazamiento horizontal a modo de «ventana» de cuarenta columnas sobre las definidas en el formato. Independientemente de esto, se determina mediante un comando específico la anchura que deberá adoptar el texto en el momento de la impresión.

Funciones básicas de edición

Una vez cumplimentada la pantalla de presentación el programa da paso al modo edición. Esta nueva pantalla está compuesta a su vez por una primera línea de «estado» que informa permanentemente del número de fila y columna en que se encuentra posicionado el cursor, conexión/desconexión de mayúsculas y demás mensajes del programa, quedando el resto de las líneas disponibles para la creación y modificación del texto.

El desplazamiento por la pantalla se realiza con ayuda de las teclas de movimiento del cursor, produciéndose un **scroll** vertical, hacia arriba o hacia abajo, al rebasarse los límites de la pantalla de edición. Además, por pulsación de **HOME** el cursor se posiciona en el ángulo superior izquierdo de la pantalla, y con **CLR HOME** en el comienzo del texto.

Así mismo, de forma similar a su empleo en la edición de programas BASIC, las teclas **DEL** e **INST** facilitan la corrección de posibles fallos tipográficos con la supresión de errores y posible inserción de nuevos caracteres dentro de una palabra escrita. La pulsación de la tecla **RETURN** tiene, sin embargo, el significado de marca de fin de párrafo, además de producir el retorno de carro y avance de línea habituales. Por ello, si en algún momento deseamos producir estos efectos secundarios sin que se produzca una marca de párrafo, podemos optar por teclear **SHIFT RETURN.**

En cualquier momento es posible alterar los colores básicos elegidos para la escritura, el fondo y el marco de la pantalla. Para ello, basta con pulsar **CTRL 1, CTRL 2** y **CTRL 3,** respectivamente; cambiando los colores en orden a su código uno a uno. Además, es posible volver a la pantalla de presentación inicial con la pulsación simultánea de **RUN STOP** y **RESTORE.**

Las teclas de función

El programa acepta la pulsación de las teclas de función **F1** a **F7** para ejecutar diferentes acciones. Sin duda, la más importante de todas ellas es **F1**, ya que da paso a un sin número de actuaciones en combinación con la pulsación de algunas teclas alfabéticas.

F2 permite la selección de nombres de ficheros a ser leídos o grabados en soporte magnético. **F3** sirve de carácter de control para la introducción de «palabras clave» de formato de impresión del texto. **F4** da paso a la pantalla de control del soporte magnético, cinta o disco. **F5** sirve de conmutador mayúsculas/minúsculas. **F6** para activar/desactivar la tabulación decimal. Por último, **F7** efectúa el salto a la siguiente posición de tabulación definida en la línea.

Las líneas de formato

La tecla de función **F7** imprime en pantalla un asterisco en vídeo inverso. Es precisamente este carácter el que indica al procesador que la secuencia de caracteres que sigue debe tratarse bien como comentario, si comienza por la secuencia de caracteres **NB**, o como descripción de formato en el resto de los casos.

En las líneas de formato pueden codificarse las características de impresión como son: columna de margen izquierdo y derecho, espaciado entre líneas, justificación, centrado automático, número de líneas a imprimir por página, definición de cabeceras y pies de página, etc.

La posibilidad de justificación a la derecha permite una inmejorable presentación de los textos. Puede combinarse además con la «indentación» o adentrado de comienzo de párrafos. Cabe también la posibilidad de ajustar a la derecha únicamente algunos bloques de texto, así como la de establecer diferentes márgenes dentro del mismo escrito.

Otra facilidad importante que ofrece **EASY SCRIPT** es la de definir líneas de cabecera que se imprimen automáticamente al comienzo de cada nueva página. Dentro de estas cabeceras puede ajustarse el texto a la izquierda, derecha o centro, incluso combinando los tres extremos. Además, es posible establecer la numeración automática de páginas a partir del límite deseado. Aunque de menor utilidad, pueden definirse igualmente pies fijos de página, con literales ajustados a los márgenes o centrados en la línea.

Existe otro tipo de línea de formato cuya misión es la de ahorrar espacio a la hora del almacenamiento del texto. Puede codificarse de esta forma el espaciado doble dentro de un texto, forzar un salto a la próxima página antes de rebasar el número máximo de líneas asignado a cada hoja, etc.

En cualquier caso, la ausencia de líneas de formato dentro de un texto no implica que la impresión de éste se produzca de una forma anárquica. Para ello, el programa establece por «defecto» determinados valores considerados como los más usuales para estos parámetros. De esta forma, en la mayoría de los casos, sólo es necesario especificar algún que otro parámetro.

Manejo del texto por bloques

Aunque el programa dispone de la posibilidad de borrado e inserción de caracteres, este procedimiento resulta lento cuando la modificación va más allá de algunas cuantas palabras. Para ello, se dispone de un proce-



dimiento de inserción y borrado de líneas completas.

Además, y como extensión de la función **F1**, pueden delimitarse bloques de texto con ayuda de las teclas de desplazamiento del cursor, con intención de desplazarlos desde la posición que ocupan a otra diferente, duplicar párrafos en otro punto del texto, o suprimir un determinado grupo de líneas no necesarias. En definitiva, podemos denominar a estas funciones de «desplazamiento», «copia» y «borrado» de párrafos de texto.

Otro tratamiento que brinda el programa es el de búsqueda, con o sin reemplazo, de determinada palabra o frase. Este método es especialmente útil cuando tratamos de eliminar la excesiva repetición de determinada palabra dentro de un texto, o cuando simplemente queremos localizar cierta zona del mismo cuando trabajamos con gran número de líneas.

Desplazamiento rápido por el texto

Como complemento del movimiento del cursor, es posible dirigirse directamente al principio o final del texto con empleo de un comando. Además, para facilitar la visualización del mismo, la pantalla puede desplazarse automáticamente hacia arriba o abajo a dos velocidades, o avanzar por páginas según nuestro deseo.

Modo disco o casete

A través de la tecla de función **F4** se tiene acceso a un subprograma de tratamiento del soporte magnético sobre el que trabajamos, cinta o disco según se seleccionó al inicio del programa. En el caso del disco, el más completo, puede ejecutarse desde el programa cualquier acción propia del Sistema Operativo de Discos (**D.O.S.**) de carga, grabación, borrado, cambio de nombre u obtención de directorios.

Visualización e impresión

Dado que la pantalla de edición no muestra las cosas en la forma exacta en que se reproducen en el papel, el programa permite efectuar una simulación en pantalla del efecto de las líneas de formato incluidas en el texto.

Así, es posible efectuar las oportunas correcciones antes de la impresión, la cual puede realizarse sobre papel contínuo o en hojas sueltas, si la impresora lo permite, emitiéndose en este caso un mensaje de aviso al final de cada página.

Edición de cartas personalizadas

Una de las características más relevantes de **EASY SCRIPT** es la de facilitar la edición de cartas personalizadas. Para ello, se basa en la peculiaridad de tratar ficheros que superan ampliamente la capacidad de memoria del ordenador, almacenándolos como diferentes bloques en disco que pueden encadenarse a la hora de la impresión para producir un único documento resultado.

Como aplicación de esta técnica, es posible confeccionar un texto base a enviar a un determinado número de clientes, por ejemplo, e incluir en el mismo antes de cada impresión algunos datos personales como el nombre, la dirección, etc., técnica denominada comunmente como **mail merge.**

DE SABIOS ES EQUIVOCARSE

Dice el refrán que de sabios es equivocarse, y quizá encontrándose este deseo de sabiduría excesivamente presente en nuestro ánimo, hemos quedado como auténticos «Einstein» al cometer no sólo una equivocación, sino dos.

Y para continuar por el camino de los dichos, «el hombre» es el único animal que tropieza dos veces en la misma piedra», y efectivamente, ni cortos ni perezosos, las dos equivocaciones han sido del mismo tipo.

Algún individuo con tendencias taxidermistas, y especialmente hábil con el bisturí, seccionó nuestros voluminosos listados de los programas ARCO y PILOTO, aparecidos en los números 17 y 18 de la anterior etapa semanal de TU MICRO COMMODORE.

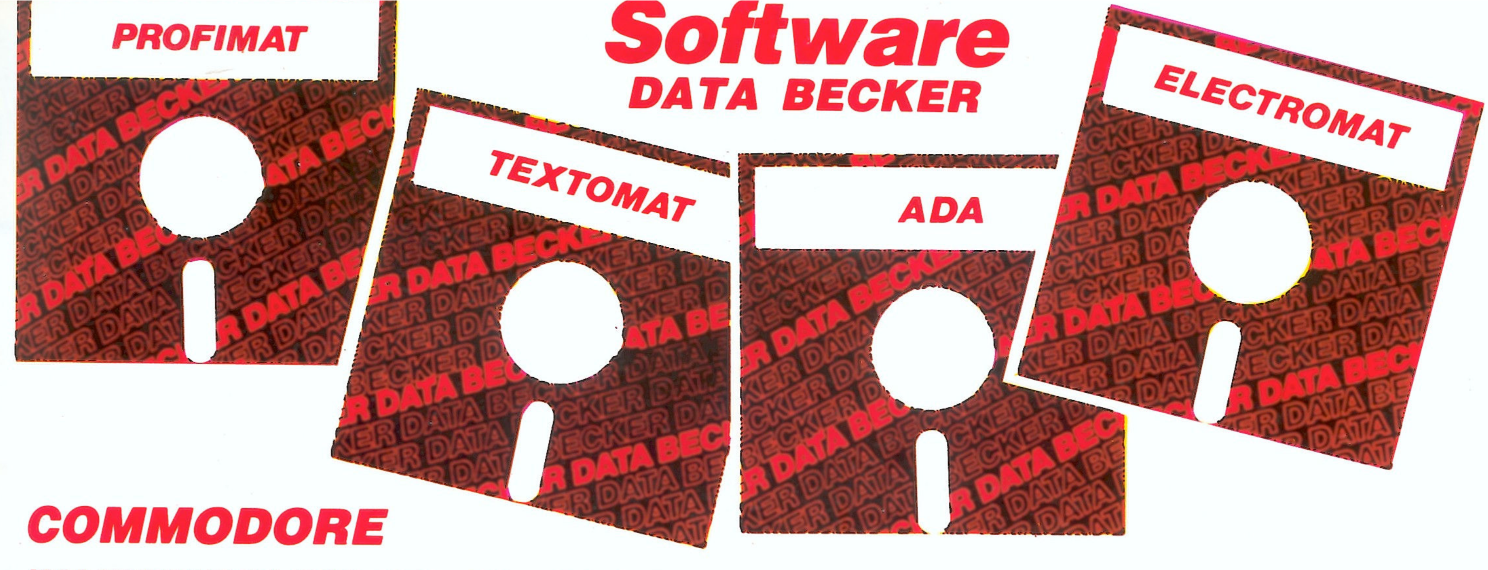
Para subsanar este problema, ahí van los dos retales. Confíamos en que hagáis una buena labor de sutura...

LISTADO ARCO

```
2350 DATA31,255,252,15,255,248,7,255,240,3,255,224,1,
255,192,0,255,0                                    -094-
2360 DATA0,0,128,0,0,128,0,255,128,0,255,144,0,255,14
4                                                  -148-
2370 DATA0,255,144,0,255,156,0,255,156,0,255
                                                   -191-
2380 DATA156,0,255,158,0,255,158,0,255,158,0,255,156,
0,255,156,0                                        -161-
2390 DATA255,156,0,255,144,0,255,144,0,255,144,0,255,
128,0,0,128,0,0,128                                -016-
2400 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,129,12
8,0,195,0,0,102                                    -214-
2410 DATA0,0,60,0,0,24,0,0,60,0,0,102,0,0,195,0,1,129
,128,0,0,0,0,0,0,0,0                               -213-
2420 DATA0,0,0,0,0,0,0,0,0                         -110-
2430 DATA0,0,112,0,0,240,0,1,144,0,3,16,0,6,16,0,12,1
6,0,12,16,0,12,16,0,12,16                          -222-
2440 DATA0,28,16,0,28,16,0,28,16,0,12,16,0,12,16,0,12
,16,0,12,16,0,6,16,0,3                             -107-
2450 DATA16,0,1,144,0,0,240,0,0,112                -220-
2460 DATA0,0,0,15,131,1,13,131,3,13,195,6,29,195,12,2
4,195,24,48,99,48,48,99                            -239-
2470 DATA96,48,99,192,96,51,128,96,51,128,96,51,128,4
8,99,192,48,99,96,48,99                            -085-
2480 DATA48,24,195,24,24,195,12,24,195,6,13,131,3,15,
131,1,15,128,0                                     -035-
2490 DATA0,0,0                                     -241-
2500 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,24,6,102
,48,12,204,127,255,254                             -062-
2510 DATA255,255,253,127,255,254,48,12,204,24,6,102,0
,0,0,0,0,0,0,0,0,0,0                               -109-
2520 DATA0,0,0,0,0,0,0,0                           -019-
2530 DATA0,63,0,0,255,192,3,255,240,3,255,240,15,255,
240,63,255,240,255,255                             -174-
2540 DATA240,2,170,160,2,106,96,10,106,104,10,170,168
,10,153,168,2,166,160                              -112-
2550 DATA2,170,160,2,234,224                       -166-
2560 DATA2,187,160,2,191,160,0,170,128,0,42,0,0,42,0,
0,42,0                                             -117-
2570 DATA0,63,0,0,255,192,3,255,240,3,255,240,15,255,
240,63,255,240,255,255                             -178-
2580 DATA240,1,85,80,2,169,80,1,169,80,1,169,80,2,169
,80,10,169,80,42,169,80                            -254-
2590 DATA10,169,80,2,169,80,2,185,80,3,234,80,2,170,1
44,2,170,160,0,10,160                              -117-
2600 DATA0,0,0,2,0,64,1,0,128,0,129,0,96,0,6,16,60,8,
0,126,0,0,255,0,6,255,96                           -222-
2610 DATA247,235,239,5,207,160,4,239,32,6,239,96,3,25
5,192,17,255,136,96,255,6                          -113-
2620 DATA0,255,0,0,60,0,4,60,32,8,255,16,16,255,8
                                                   -163-
2630 POKE54296,15:POKE54276,65:POKE54277,144
                                                   -245-
2640 POKE54278,243:POKE54273,200:POKE54272,200
                                                   -065-
2650 POKE54276,16:RETURN                           -209-
2660 FORL=54272TO54296:POKEL,0:NEXT                -153-
2670 POKE54276,33:POKE54296,15:POKE54277,128:POKE5427
8,128:POKE54272,200                                -091-
2680 POKE54273,200:POKE54276,0:RETURN              -201-
2690 PRINT"{HOM}":PRINTTAB(11)"{11 ABJ}{BLK}G A M E[4
 ESP]O V E R"                                      -075-
2700 RESTORE:GOSUB50:PRINT"{CLR}":POKEV+21,0:RETURN
                                                   -141-
2710 PRINT"{HOM}{YEL}{9 ABJ}{DCH}LO SIENTO, NO HAS SU
PERADO LA PRUEBA"                                  -163-
2720 FORT=0TO16:FORVV=1TO50:NEXTVV:POKEV+42,T:NEXTT:I
FPL=1THENPOKEV+42,10                               -071-
2730 PRINT"{ARB}[38 ESP]":RETURN                   -040-
2740 FORDS=1TO3:Z=Z-8:POKEV+14,Z:FORT=1TO30:NEXTT:NEX
TDS:POKEV+14,0:RETURN                              -159-
2750 LETLP=0:RESTORE:GOSUB50                       -089-
2760 POKEV+40,4:POKEV+28,40:POKEV+23,68:POKEV+29,84
                                                   -029-
2770 FORSW=12352TO12414:LETLP=LP+1                 -074-
2780 FJ(LP)=PEEK(SW):READWS:POKESW,WS              -180-
2790 NEXTSW:LETGA=1:RETURN                         -010-
2800 LETUI=Y+11:IFABS(UI-I)>12THENRETURN           -068-
2810 IFA<4ORA>6THENRETURN                          -144-
2820 LETP=750:GOTO1340                             -053-
2830 SYS64738                                      -248-
```

LISTADO PILOTO

```
30002 DATA150,0,2,190,128,2,150                    -044-
30003 DATA128,0,190,0,0,150,0                      -195-
30004 DATA0,190,0,2,150,128,2                      -200-
30005 DATA215,128,2,150,128,2,215                  -154-
30006 DATA128,42,215,168,145,150,70                -010-
30007 DATA132,150,18,145,190,70,132                -002-
30008 DATA170,18,33,190,72,26,85                   -124-
30009 DATA164,16,20,4,84,85,21                     -018-
30010 REM *** ***                                  -052-
40001 DATA40,0,40,182,0,158,158                    -054-
40002 DATA0,182,182,255,158,10,255                 -213-
40003 DATA160,15,0,240,245,85,95                   -114-
40004 DATA192,0,3,245,85,95,60                     -021-
40005 DATA68,124,46,190,184,8,0                    -073-
40006 DATA32,44,0,56,170,0,170                     -004-
40007 DATA0,0,0,0,0,0,0                            -141-
40008 DATA0,0,0,0,0,0,0                            -142-
40009 DATA0,0,0,0,0,0,0                            -143-
40010 REM *** ***                                  -053-
50001 DATA0,0,0,0,0,0,0                            -136-
50002 DATA0,0,0,0,0,0,0                            -137-
50003 DATA0,0,0,0,0,0,0                            -138-
50004 DATA0,0,0,170,170,170,170                    -043-
50005 DATA170,170,149,85,86,149,85                 -240-
50006 DATA86,149,85,86,170,170,170                 -236-
50007 DATA170,170,170,0,0,0,0                      -198-
50008 DATA0,0,0,0,0,0,0                            -143-
50009 DATA0,0,0,0,0,0,0                            -144-
50010 REM *** ***                                  -054-
60000 DATA34,75,38,126,38,126,34,75,38,126,38,126,40,
200,43,52,45,198                                   -161-
60001 DATA34,75,36,85,38,126,40,200,43,52,45,198,48,1
27                                                 -239-
60002 DATA 51,97,51,97,54,111,51,97,51,97,68,149,72,1
69                                                 -247-
60003 DATA 64,188,61,126,51,97,51,97,54,111,51,97,51,
97                                                 -238-
60004 DATA48,127,45,198,43,52,40,200,38,126,36,85,34,
75                                                 -242-
60005 DATA48,127,45,198,43,52,40,200,38,126,36,85,34,
75                                                 -243-
61000 DATA238,3,208,238,3,208,238,3,208,238,11,208,23
8,11,208,238,11,208,96                             -193-
```



COMMODORE

KALKUMAT 11.800 ptas. *(Disponible final de junio)*

El KALKUMAT es una hoja de cálculo electrónica, que puede aplicarse para efectuar cualquier tipo de cálculos, estadísticas, proyectos de planings y simulaciones.

TEXTOMAT 5.850 ptas

¡El procesador de textos más vendido en Alemania, ahora también disponible para **AMSTRAD**

PROFIMAT 5.850 ptas.

Quien quiere acceder a la vida interior del computador, necesita disponer de herramientas especiales. Aquí es donde PROFIMAT entra en acción, ofreciendo un confortable monitor de código máquina así como un macroensamblador.

ELECTROMAT 4.100 ptas.

Con ELECTROMAT, la creación de esquemas de circuitos electrónicos se convierte en una diversión —todos los esquemas pueden editarse directamente en pantalla.

ADA — curso de entrenamiento 11.700 ptas.

Este lenguaje de programación del futuro, el cual al igual que COBOL fue encargado al pentágono, ahora puede ser aprendido por el usuario del C-64, mediante el curso de entrenamiento de DATA BECKER. El curso de entrenamiento de ADA además contiene un compilador que ofrece un exhaustivo SUBSET, y los elementos principales de este lenguaje.

PLATINE 64 - LA LLAVE PARA EL MERCADO ELECTRONICO 30.000 ptas.

Hasta ahora los sistemas de desarrollo de placas de circuito impreso (platinas) sólo eran accesibles con mucho dinero y a base de computadoras caras. De allí que había muchos, que necesitaban un sistema de este tipo, pero pocos que podían costeárselo. Esto ha llegado a su fin. Compare los datos, PLATINE 64 a este precio y en esta categoría de ordenadores, es una auténtica sensación a nivel mundial.*

128 BASIC — 7.500 ptas.

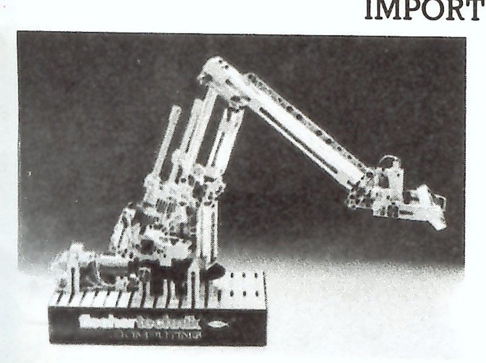
¡Por fin! Un potente compilador Basic para 128. Programas Basic de 4 a 30 veces más rápidos. Cómoda estructuración tipo MENU.

fischertechnik®

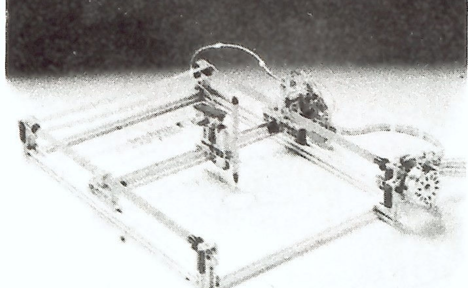


ROBOTS DIDACTICOS

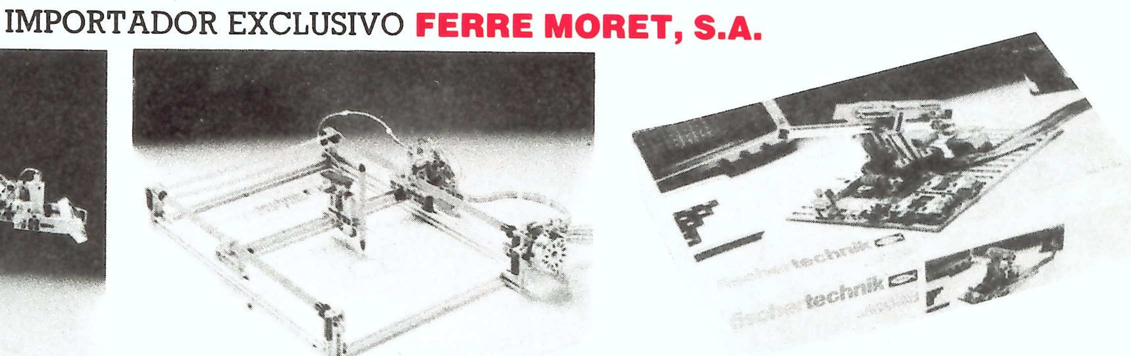
IMPORTADOR EXCLUSIVO **FERRE MORET, S.A.**



Art. 30.572 Trainingsroboter



Art. 30.571 Plotter-Scanner



Art. 30.554 Computing

Solicite información:
FERRE MORET, S.A. C/ Tuset, 8, ent.° 2.° - Tel. 217 63 13 - 217 62 38 - 08006 BARCELONA



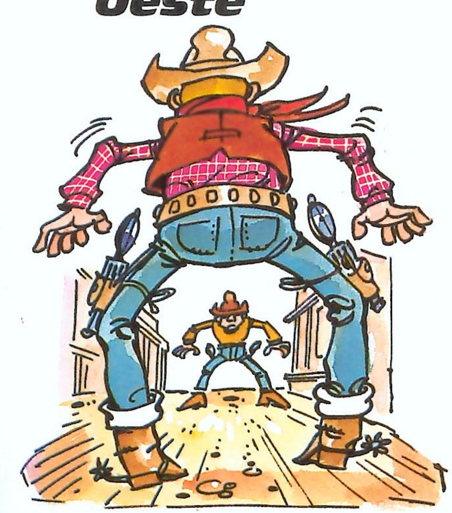
1. Back to the future

Los problemas y oportunidades que representaría al hombre poder volver al pasado serían de diversa índole. Quizás la más significativa consistiría en poder cambiar el curso de la historia, aunque hoy por hoy no sabemos si sería posible, ya que se podrían plantear multitud de paradojas, como la que se nos plantea en «Back to the Future».



Colaboremos todos en la confección de estos 7 MAGNIFICOS de TU MICRO COMMODORE. Envíanos el nombre de tu programa favorito, dejando muy claro tu nombre y dirección. Todos los meses, sortearemos cinco suscripciones por un año a nuestra revista entre las respuestas recibidas. Anímate y escríbenos a: TU MICRO COMMODORE. (7 MAGNIFICOS). Apartado de Correos 61.294. 28080 MADRID.

2. La ley del Oeste



3. Rambo

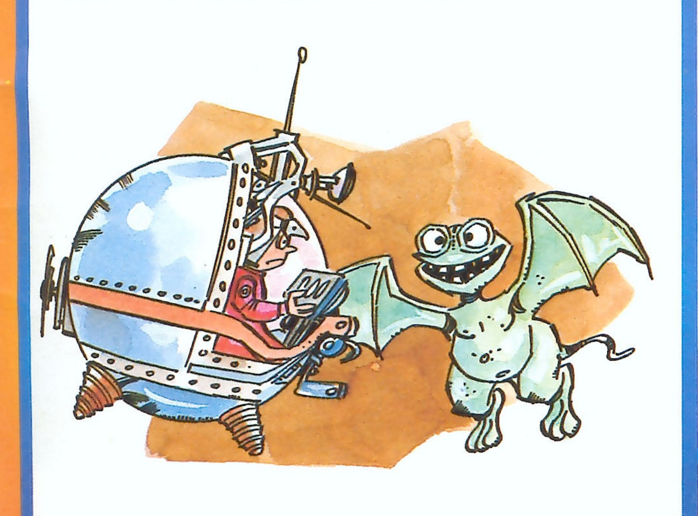


4. Commando



5. Little Computer People

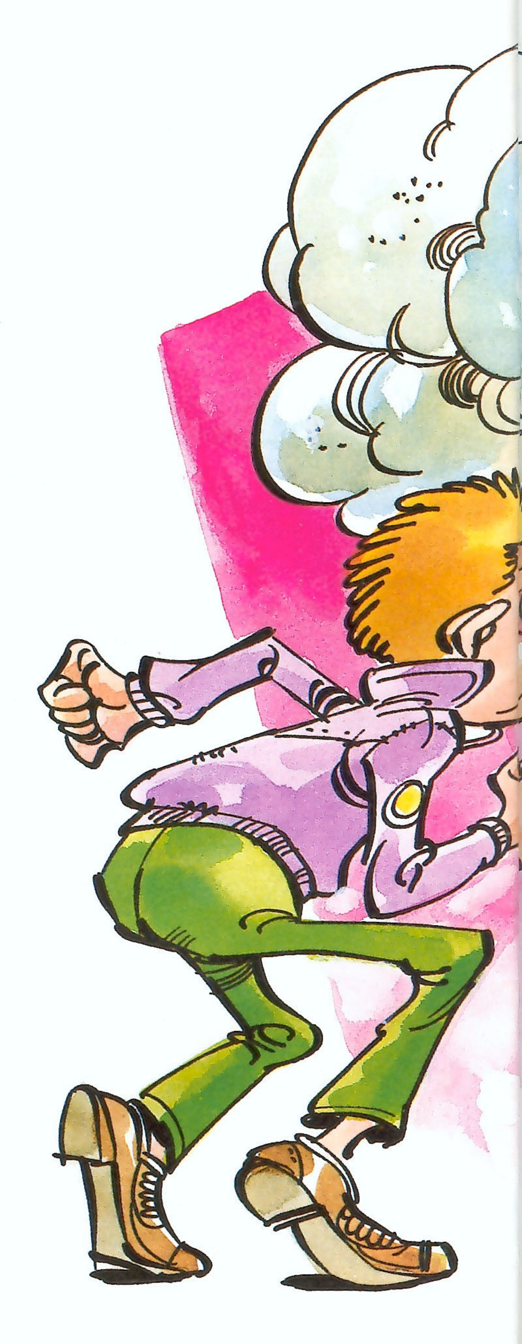
6. The Eidolon

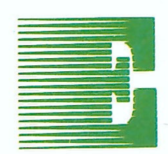


7. Hardball

Fumar

¿Fumar o no fumar?, that is the question. ¿Enfermar poco a poco con el humo del tabaco que consumimos nosotros mismos, o con el que consumen los demás?, that sigue siendo la question. No obstante, fumadores y no fumadores se enfrentan diariamente unos contra otros por una futesa, al fin y al cabo puro humo (aunque eso sí, indeseable y perjudicial para la salud).





El reloj de la pared acababa de marcar las diez, la hora propicia para proceder a encender un nuevo cigarrillo. En el fondo me dolía hacerlo, pues antes debía apagar el que ya me estaba fumando, y ni siquiera recordaba un solo instante de mi vida sin un cigarrilo entre mis manos. Esto era así desde hace varios años, cuando empecé a contraer el insano vicio del tabaco, aprovechando después cualquier excusa con el fin de aumentar las dosis de nicotina, sobre todo cuando empleaba mi tiempo en programar a mi querido ordenador: el Commodore 64. Incluso ahora, hay quien dice que tengo cara de cigarrillo encendido...

Día tras día, noche tras noche, las resentidas teclas del Commodore me suplicaban entre «clicks» que fueran un poco más suave, menos brusco, y que, por favor, no las envolviera tras las volutas de humo de mi tabaco. Por contra, yo nunca he sido muy considerado, y menos con una máquina de circuitos de silicio cuya diversión predilecta es siempre la misma: tratar de obstaculizar al máximo mi tarea como programador. Así, ni corto ni perezoso, me puse los guantes de teclear, la chistera de teclear, la pajarita de teclear y... ¡zass! Unos pocos días es todo cuanto me hizo falta para poner a punto el programa FUMAR.

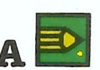
Realmente, la parte más difícil fue la del diseño de sprites, los cuales debían representar fielmente ciertos «items» (un hombre, un mechero, un cigarrillo y una nube de humo), y añadir además varias posiciones para poder efectuar un movimiento animado. Quizás haya algún otro aspecto interesante como las interrupciones o la música, si bien el resto del juego carece de sentido lógico hasta tal punto que pudiera haberse llamado EL VERDULERO LOCO, en caso de haberse cambiado la forma de los sprites, sustituyéndolos por productos de la huerta.

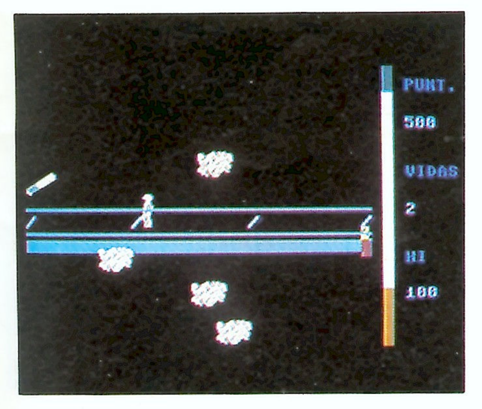
Referente a la utilización del programa, cabe destacar la necesidad de conectar un joystick en el port 2, el cual será presionado durante la pantalla de presentación para acceder así a otras secciones del programa. En esta línea, el primer obstáculo a superar consiste en coger tres cigarrillos, evitando las nubes, y cambiando de nivel mediante el mechero. Nunca debemos subir a un piso superior si no hemos cogido el cigarrillo del piso anterior. Aquí, y a lo largo de toda la sesión de juego, la barra de espacio produce una pausa, **F1** establece una velocidad lenta para nuestro hombre, **F3** duplica dicha velocidad, **F5** conecta la música (¡Noooo!) y **F7** la desconecta (¡Yupiiiii!).

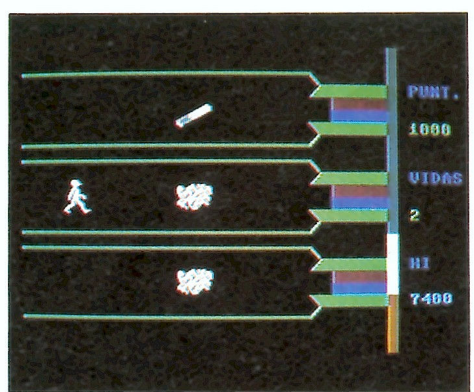
En la segunda pantalla podremos obtener más puntos cuantos más objetos golpeemos con nuestro cuerpo, y conseguir una bonificación especial de 5.000 puntos, si logramos provocar la desaparición de todos los objetos situados alrededor nuestro. Por último, se trata de ver cuan rápidos son nuestros reflejos, pues habremos de escoger el pasillo adecuado, aquél por el cual discurre el cigarrillo para así colisionar con él. Recuerda que el contacto con una nube supone una penalización de dos puntos.

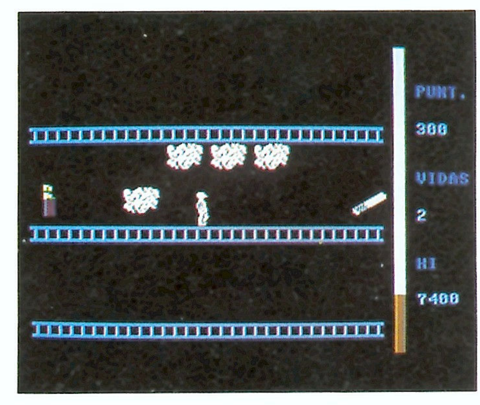
Antes de empezar a teclear recordad que este programa no pretende ejercer ninguna influencia sobre vuestro estado de ánimo, tanto si sois fumadores como si no.






PUNT.
500
VIDAS
2
HI
100


PUNT.
1000
VIDAS
2
HI
7400


PUNT.
300
VIDAS
2
HI
7400

LISTADO

```
10 POKE789,234                                              -029-
20 FORN=15360TO15999:READA:POKEN,A;NEXT                     -219-
30 FORN=49201TO49325:READA:POKEN,A:NEXT                     -211-
40 FORN=0TO93:READA:POKE52922+N,A:NEXT                      -152-
50 POKE2,0:POKE252,0:POKE253,0                              -255-
60 POKE53276,2:POKE53285,8:POKE53286,7                      -194-
70 FORN=0TO7:READA:POKE53287+N,A:NEXT                       -107-
80 FORN=0TO7:READA:POKE2040+N,A:NEXT                        -041-
90 POKE251,1                                                -173-
95 FORN=50000TO50036:READA:POKEN,A:NEXTN                    -024-
97 FORN=0TO29:READA:POKE50993+N,A:NEXT                      -169-
100 POKE789,234:FORL=54272TO54295:POKEL,0:NEXT:POKE54
296,15:VI=3                                                 -127-
105 POKE49306,234:POKE49316,234:POKE53269,0:HI=PEEK(2
51)*100                                                     -132-
107 FORN=0TO3:POKE2044+N,249:NEXT                           -032-
110 POKE54277,36:POKE54278,240                              -180-
120 POKE49289,76:POKE49290,49:PU=0                          -225-
200 POKE53280,0:POKE53281,0:PRINTCHR$(142),CHR$(8)
                                                            -109-
210 PRINT"{CLR}[10 ESP]{BLU} {RON}{NRJ}[4 ESP]{WHT}[1
1 ESP]{GR1}[2 ESP]{ROF}{ACL}[12 ESP]";                      -204-
220 PRINT"{2 ABJ}{BLU}[5 ESP]{BLU} {RON}[4 ESP]{ROF}[
2 ESP]{RON}{BLU} {ROF}[2 ESP]{RON}{BLU} {ROF}[2 ESP]{
RON}{BLU} {ROF}[3 ESP]{RON}{BLU} {ROF}[2 ESP]{RON}{BL
U}[4 ESP]{ROF}[2 ESP]{RON}{BLU}[3 ESP]{ROF}"
                                                            -091-
230 PRINT"{BLU}[5 ESP]{RON}{NRJ}[4 ESP]{ROF}[2 ESP]{R
ON}{NRJ} {BLU} {ROF} {RON}{NRJ} {BLU} {ROF} {RON}{NRJ
} {BLU}[2 ESP]{ROF} {RON}{NRJ} {BLU} {ROF} {RON}{NRJ}
[4 ESP]{BLU} {ROF} {RON}{NRJ}[3 ESP]";                      -012-
240 PRINT"{ROF}{BLU} {RON} {ROF}[4 ESP]{BLU} ";
                                                            -022-
250 PRINT"{BLU}[5 ESP]{RON}{NRJ} {BLU} {ROF}[4 ESP]{R
ON}{NRJ} {BLU} {ROF} {RON}{NRJ} {BLU} {ROF} {RON}{NRJ
}[2 ESP]{ROF}{BLU} {RON}{NRJ}[2 ESP]{BLU} {ROF} {RON}
{NRJ} {BLU} {ROF} {RON}{NRJ} {BLU} {ROF} ";                 -126-
260 PRINT"{RON}{NRJ} {BLU} {ROF} {RON}{NRJ} {BLU} {RO
F}[4 ESP]{BLU} ";                                           -171-
270 PRINT"{BLU}[5 ESP]{RON}{NRJ} {BLU}[2 ESP]{ROF}{NR
J}[3 ESP]{RON} {BLU} {ROF} {RON}{NRJ} {BLU} {ROF} {RO
N}{NRJ} {BLU} {NRJ} {ROF}{BLU} {RON}{NRJ} {BLU} {ROF}
 {RON}{NRJ} {BLU}[2 ESP]{NRJ} {BLU} {ROF} {RON}{NRJ}
";                                                          -047-
280 PRINT"{BLU}[2 ESP]{NRJ} {ROF}[5 ESP]{BLU}"
                                                            -220-
290 PRINT"{BLU}[5 ESP]{RON}{NRJ}[2 ESP]{ROF}[4 ESP]{R
ON}{NRJ} {BLU} {ROF} {RON}{NRJ} {BLU} {ROF} {RON}{NRJ
} {BLU} {ROF}[2 ESP]{RON}{NRJ} {BLU} {ROF} {RON}{NRJ}
[4 ESP]{BLU} {ROF} {RON}{NRJ}[2 ESP]";                      -242-
300 PRINT" {ROF}{BLU} {RON} {ROF}[4 ESP]{BLU}"
                                                            -216-
310 PRINT"{BLU}[5 ESP]{RON}{NRJ} {BLU} {ROF}[4 ESP]{R
ON}{NRJ} {BLU}[2 ESP]{NRJ} {BLU} {ROF} {RON}{NRJ} {BL
U} {ROF}[2 ESP]{RON}{NRJ} {BLU} {ROF} {RON}{NRJ} {BLU
} {ROF}{NRJ} {RON} {BLU} {ROF} {RON}{NRJ} ";
                                                            -138-
320 PRINT"{BLU} {ROF} {RON}{NRJ} {BLU} {ROF}"
                                                            -251-
330 PRINT"[5 ESP]{RON}{NRJ} {ROF}[5 ESP]{RON}{NRJ}[4
ESP]{ROF}[2 ESP]{RON}{NRJ} {ROF}[3 ESP]{RON}{NRJ} {RO
F}[2 ESP]{RON}{NRJ} {ROF}[2 ESP]{RON}{NRJ} {ROF}[2 ES
P]{RON}{NRJ} {ROF}[2 ESP]{RON}{NRJ} ";                      -152-
340 PRINT"{ROF}[5 ESP]{BLU}"                                -217-
350 PRINT"{9 ABJ}{ACL}[5 ESP]{PUR} MUSICA, GRAFICOS Y
 PROGRAMA{ACL}[7 ESP]";                                    -027-
360 PRINT"{ABJ}{ACL}[7 ESP]{RED}[2 ESP]|U||19 *||I|{A
CL}[10 ESP]";                                               -100-
370 PRINT"{CYN}[9 ESP]{RED}|G|{CYN}PABLO GARCIA MOLIN
A{RED}|H|{CYN}[9 ESP]{RED} ";                               -251-
380 PRINT"{RED}[9 ESP]|J||19 *||K|{HOM}"                    -176-
390 POKE53250,250:POKE53251,154:POKE53249,156
                                                            -020-
400 POKE53249,156:FORN=0TO3:POKE53257+(2*N),150:POKE5
3256+(2*N),40+(30*N):NEXT                                   -175-
410 POKE2040,240:POKE53269,247:POKE789,192:FORN=165TO
230:POKE53248,N                                             -233-
415 IFPEEK(56320)=111THEN600                                -084-
420 IFN/5<>INT(N/5)THEN450                                  -039-
430 IFPEEK(2040)=242THENPOKE2040,240:GOTO450
                                                            -073-
440 POKE2040,PEEK(2040)+1                                   -113-
450 NEXTN                                                   -070-
460 FORN=0TO100:NEXTN:POKE2040,243:FORN=230TO165STEP-
1:POKE53248,N                                               -078-
465 IFPEEK(56320)=111THEN600                                -089-
470 IFN/5<>INT(N/5)THEN500                                  -040-
480 IFPEEK(2040)=245THENPOKE2040,243:GOTO500
                                                            -080-
490 POKE2040,PEEK(2040)+1                                   -118-
500 NEXTN:GOTO400                                           -073-
600 POKE789,234:POKE54276,16:FORN=2044TO2047:POKEN,24
8:FORV=0TO500:NEXTV,N                                       -034-
610 POKE53269,0:POKE253,0                                   -170-
630 GOSUB19100                                              -052-
640 PRINT"{HOM}{6 ABJ}<31 R>"                               -146-
650 PRINT"<31 E>"                                           -251-
660 PRINT"{6 ABJ}<31 R>"                                    -129-
670 PRINT"<31 E>"                                           -253-
680 PRINT"{6 ABJ}<31 R>"                                    -131-
690 PRINT"<31 E>{HOM}":C=3:A=0                              -231-
700 H=0:POKE53248,56:POKE53249,207-A:POKE53250,20:POK
E53251,202-A                                                -162-
710 POKE53254,250:POKE53255,200-A:FORN=0TO3:POKE53256
+2*N,90+30*N                                                -214-
720 POKE53257+2*N,173-A:NEXT:POKE53269,255:POKE49289,
234:POKE49290,234                                           -031-
730 POKE789,192                                             -089-
740 R=INT(RND(1)*4)*2:FORZ=173-ATO209-ASTEP7:POKE5325
7+R,Z                                                       -248-
750 GOSUB19000:GOSUB19700:GOSUB19600:GOSUB19500:ON H
GOTO760,700,600                                             -030-
755 NEXTZ:POKE53257+R,173-A:GOTO740                         -137-
760 IFC<>0THENGOSUB19800:GOTO600                            -243-
770 POKE53269,0:POKE789,234:POKE54276,16:GOSUB19100
                                                            -225-
773 PRINT"{HOM}{13 ABJ}|31 *|"                              -194-
775 PRINT"|N|{9 DCH}|N|{9 DCH}|N|{9 DCH}|N|"
                                                            -219-
777 PRINT"|31 *| "                                          -246-
778 PRINT"{RON}[31 ESP]{ROF}"                               -027-
780 H=0:POKE53248,148:POKE53249,148:POKE53250,250:POK
E53251,108                                                  -062-
790 POKE53254,25:POKE53255,188                              -191-
800 FORN=0TO3:POKE53256+(2*N),INT(RND(1)*150)+50:NEXT
N                                                           -169-
820 POKE53269,255:POKE789,192                               -147-
825 FORN=0TO19:POKE55328+(40*N),11                          -196-
827 FORN1=0TO3                                              -106-
830 SYS50000:GOSUB19000:SYS50000:GOSUB19600:SYS50000
                                                            -121-
840 P=PEEK(53278)AND1:P=PEEK(53278)AND1:IFP=1THENSYS5
0000:GOSUB1000                                              -059-
850 NEXTN1,N                                                -245-
860 FORN=0TO19:POKE55328+(40*N),0                           -145-
865 FORN1=0TO3                                              -108-
870 SYS50000:GOSUB19000:SYS50000:GOSUB19600:SYS50000
                                                            -125-
880 P=PEEK(53278)AND1:P=PEEK(53278)AND1:IFP=1THENSYS5
0000:GOSUB1000                                              -063-
890 NEXTN1,N                                                -249-
893 PRINT"{HOM}"                                            -168-
895 IFPEEK(53269)=1ORPEEK(53269)=5THENPU=PU+5000:VI=V
I+1:GOSUB19210                                              -066-
900 POKE789,234:POKE54276,16:GOSUB19700                     -020-
910 GOTO2000                                                -181-
1000 FORZ=2TO7:SYS5000Q:P=PEEK(53278)ANDZ                   -012-
1010 IFPEEK(53278)AND(2^Z)=(2^Z)THENPOKE53269,ABS(PEE
K(53269)-(2^Z)):PU=PU+100                                   -148-
1020 NEXTZ:RETURN                                           -150-
2000 K=11:POKE53275,105:GOSUB19100:POKE49289,76:POKE4
9290,49:POKE2040,240                                        -191-
2005 FORN=1TO3                                              -096-
2010 PRINT"{ABJ}{GRN}<25 Y>|M|"                             -143-
2020 PRINTTAB(25)"{RON}<RON>[6 ESP]{ROF}"                   -081-
2030 PRINTTAB(27)"{RED}{RON}[5 ESP]{ROF}"                   -167-
2040 PRINTTAB(27)"{BLU}{RON}[5 ESP]{ROF}"                   -171-
2050 PRINTTAB(25)"{GRN}|GRN|{RON}[6 ESP]{ROF}"
                                                            -168-
2060 PRINT"<25 P>|N|":NEXTN:PRINT"{HOM}"                    -131-
2070 POKE53248,50:POKE53249,135:POKE53258,255:POKE532
60,255                                                      -077-
2080 POKE53254,255:POKE53269,105:POKE788,49:POKE789,1
99                                                          -161-
2090 FORW=0TO20STEP.05:POKE55328+(40*INT(W)),K
                                                            -238-
2095 IFPEEK(53258)<175THEN2140                              -203-
2100 A=INT(RND(1)*3)+1:ONAGOTO2110,2120,2130
                                                            -180-
2110 POKE53255,80:POKE53259,135:POKE53261,190:GOTO214
0                                                           -075-
2120 POKE53255,135:POKE53259,80:POKE53261,190:GOTO214
0                                                           -076-
2130 POKE53255,190:POKE53259,80:POKE53261,135
                                                            -019-
```

```
2140 IFPEEK(2040)=242THENPOKE2040,240:GOTO2160
                                                      -169-
2145 IF PEEK (53258)>35 AND PEEK(53258)<70 THEN POKE7
89,234:GOSUB18000                                     -207-
2150 POKE2040,PEEK(2040)+1                            -161-
2160 P1=PEEK(53249):P2=PEEK(56320)                    -147-
2170 IFP2=126THENIFP1<>80THENPOKE53249,P1-55:GOTO2190
                                                      -218-
2180 IFP2=125THENIFP1<>190THENPOKE53249,P1+55
                                                      -203-
2190 REM                                              -208-
2500 GOSUB19000:NEXTW                                 -049-
2510 IFK=11THENK=0:GOTO2090                           -134-
3000 POKE789,234:POKE54276,16:IFPU>HITHENPOKE251,(PU/
100)                                                  -152-
3010 FORN=0TO2000:NEXTN:POKE53269,0:RUN100            -064-
15000 END                                             -237-
18000 IFPEEK(53255)=PEEK(53249)THENPU=PU+100:GOTO 180
20                                                    -220-
18010 PU=PU-200:IFPU<0THENPU=0                        -123-
18020 POKE789,199:GOSUB19230:RETURN                   -148-
19000 PZ=PEEK(53248):GETW$:POKE198,0:IFW$=""THENRETUR
N                                                     -077-
19010 IFW$="{F1 }"THENPOKE49306,234:POKE49316,234:RET
URN                                                   -163-
19020 IFW$="{F3 }"THENPOKE53248,INT(PZ/2)*2:POKE49306
,232:POKE49316,202:RETURN                             -220-
19030 IFW$="{F5 }"THENPOKE54296,15:RETURN             -067-
19040 IFW$="{F7 }"THENPOKE54296,0:RETURN              -015-
19050 IFW$<>" "THENRETURN                             -022-
19055 IFPEEK(789)=199THEN19080                        -226-
19060 POKE789,234:POKE54276,16:FORN=0TO100:NEXT:POKE1
98,0:WAIT197,60                                       -082-
19070 POKE198,0:POKE789,192:RETURN                    -065-
19080 POKE788,49:POKE789,234:POKE54276,16:FORN=0TO100
0:NEXT:POKE198,0                                      -139-
19090 WAIT197,60:POKE198,0                            -242-
19095 POKE789,199:POKE788,49                          -119-
19097 RETURN                                          -010-
19100 PRINT"{WHT}{CLR}":FORN=1056TO2016STEP40:POKEN,1
60:NEXT                                               -251-
19200 FORN=56128TO56288STEP40:POKEN,8:NEXT            -120-
19210 PRINTTAB(34)"{BLU}{2 ABJ}PUNT.":PRINTTAB(34)"{6
 ABJ}VIDAS":PRINTTAB(34)"{6 ABJ}HI"                   -059-
19220 PRINT"{ACL}{2 ABJ}"TAB(33)HI"{HOM}"             -033-
19230 PRINT"{5 ABJ}"TAB(33)PU:PRINT"{6 ABJ}"TAB(33)VI
"{HOM}":RETURN                                        -135-
19500 P=PEEK(53278)AND1:P=PEEK(53278)AND1:IFP=0THENRE
TURN                                                  -014-
19510 P=PEEK(53248):IFP<56THEN19530                   -026-
19520 IFP>212THENC=C-1:POKE53269,243:PU=PU+100:RETURN
                                                      -241-
19525 GOTO19550                                       -099-
19530 IFA=128THENH=1:RETURN                           -201-
19540 A=A+64:H=2:RETURN                               -130-
19550 H=3:GOSUB19800:RETURN                           -178-
19600 IFPEEK(254)<>119ANDPEEK(254)<>123THENRETURN
                                                      -216-
19610 IFPEEK(254)<>119THEN19640                       -009-
19620 IFPEEK(2040)=>242THENPOKE2040,240:RETURN
                                                      -000-
19630 POKE2040,PEEK(2040)+1:RETURN                    -246-
19640 IFPEEK(2040)<243ORPEEK(2040)=245THENPOKE2040,24
3:RETURN                                              -124-
19650 POKE2040,PEEK(2040)+1:RETURN                    -248-
19700 PRINT"{HOM}{6 ABJ}"TAB(33)PU:RETURN             -184-
19710 PRINT"{HOM}{6 ABJ}"TAB(33)PU:PRINT"{7 ABJ}"TAB(
35)VI"{HOM}":RETURN                                   -193-
19800 VI=VI-1:IFVI=0THENPOKE789,234:POKE54276,16:FORN
=0TO2000:NEXT:GOTO19820                               -208-
19810 RETURN                                          -003-
19820 IFPU>HITHENHI=PU:POKE251,HI/100                 -071-
19830 RUN 100                                         -203-
20000 DATA 0,60,0,0,126,0,0,127,0,0,60,0,0,24,0,0
                                                      -085-
20010 DATA 60,0,0,124,0,0,236,0,1,174,0,3,47,192,3,51
                                                      -057-
20020 DATA 224,6,62,0,0,59,0,0,119,0,0,115,128,0,97,1
92                                                    -175-
20030 DATA 1,192,192,3,128,192,6,0,96,14,0,120,3,0,96
,0                                                    -181-
20040 DATA 0,60,0,0,126,0,0,127,0,0,60,0,0,24,0,0
                                                      -089-
20050 DATA 60,0,0,46,0,0,86,0,0,86,0,0,90,0,0,47
                                                      -069-
20060 DATA 0,0,59,0,0,62,0,0,29,0,0,27,0,0,27,128
                                                      -116-
20070 DATA 0,59,0,0,54,0,0,52,0,0,58,0,0,62,0,0
                                                      -010-
20080 DATA 0,60,0,0,126,0,0,127,0,0,60,0,0,24,0,0
                                                      -093-
20090 DATA 60,0,0,110,0,0,110,0,0,222,0,0,223,128,1,1
90                                                    -134-
20100 DATA 224,1,190,96,1,62,0,0,111,0,0,119,128,0,11
5,192                                                 -056-
20110 DATA 1,225,192,3,128,192,6,0,224,14,0,124,3,0,4
8,0                                                   -219-
20120 DATA 0,60,0,0,126,0,0,254,0,0,60,0,0,24,0,0
                                                      -089-
20130 DATA 60,0,0,60,0,0,52,0,0,116,0,3,244,0,7,204
                                                      -201-
20140 DATA 0,0,124,0,0,220,0,0,238,0,1,206,0,3,134,0
                                                      -241-
20150 DATA 3,3,128,3,1,192,6,0,96,30,0,112,6,0,192,0
                                                      -021-
20160 DATA 0,60,0,0,126,0,0,254,0,0,60,0,0,24,0,0
                                                      -093-
20170 DATA 60,0,0,116,0,0,106,0,0,106,0,0,90,0,0,244
                                                      -247-
20180 DATA 0,0,220,0,0,124,0,0,184,0,0,216,0,1,216,0
                                                      -244-
20190 DATA 0,220,0,0,108,0,0,44,0,0,92,0,0,124,0,0
                                                      -145-
20200 DATA 0,60,0,0,126,0,0,254,0,0,60,0,0,24,0,0
                                                      -088-
20210 DATA 60,0,0,118,0,0,118,0,0,123,0,1,251,0,7,125
                                                      -043-
20220 DATA 128,6,125,128,0,124,128,0,246,0,1,238,0,3,
206,0                                                 -063-
20230 DATA 3,135,128,3,1,192,7,0,96,62,0,112,12,0,192
,0                                                    -173-
20240 DATA 0,0,240,0,3,112,0,3,112,0,0,80,0,0,0,0
                                                      -082-
20250 DATA 3,192,0,3,240,0,3,240,0,3,84,0,2,160,0,2
                                                      -210-
20260 DATA 168,0,2,168,0,2,168,0,2,168,0,2,168,0,2,16
8                                                     -140-
20270 DATA 0,2,168,0,2,168,0,2,168,0,2,168,0,0,0,240
                                                      -019-
20280 DATA 0,3,192,0,3,112,0,3,112,0,1,80,0,0,0,0
                                                      -096-
20290 DATA 3,192,0,3,240,0,3,240,0,3,84,0,2,160,0,2
                                                      -214-
20300 DATA 168,0,2,168,0,2,168,0,2,168,0,2,168,0,2,16
8                                                     -135-
20310 DATA 0,2,168,0,2,168,0,2,168,0,2,168,0,0,0,0
                                                      -168-
20330 DATA 0,0,0,0,0,0,0,0,0,0,51,96,112,251,222,251
                                                      -245-
20340 DATA 252,239,251,223,255,231,239,247,255,239,18
7,126,222,252,110,249                                 -141-
20350 DATA 188,49,255,254,31,255,238,103,246,244,253,
239,248,253,223,156                                   -055-
20360 DATA 60,221,0,22,110,0,2,60,0,0,0,0,0,0,0
                                                      -082-
20370 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0         -234-
20380 DATA 0,0,0,0,0,0,0,12,0,0,63,0,0,255,0,3
                                                      -198-
20390 DATA 255,0,15,252,0,63,240,0,255,192,2,255,0,10
,188,0                                                -126-
20400 DATA 42,160,0,170,128,0,234,0,0,248,0,0,48,0,0,
0                                                     -103-
40000 DATA 234,234,234,234,234,234,234,234,234,234,16
4,252,200,192,9,240                                   -013-
40010 DATA 5,132,252,76,104,192,160,0,132,252,166,253
,189,186,206,141                                      -129-
40020 DATA 1,212,189,187,206,141,0,212,169,17,141,4,2
12,232,232,224                                        -012-
40030 DATA 92,208,2,162,0,134,253,164,2,200,192,15,24
0,5,132,2                                             -011-
40040 DATA 76,137,192,160,0,132,2,174,249,7,224,246,2
40,5,162,246                                          -191-
40050 DATA 76,134,192,162,247,142,249,7,76,49,234
                                                      -194-
40060 DATA 173,0,220,174,0,208,201,119,208,6,224,254,
240,12,232,232,201,123,208                            -076-
40070 DATA 6,224,24,240,2,202,202,142,0,208,133,254,7
6,49,234                                              -231-
50000 DATA 57,172,64,188,68,149,64,188,57,172,51,97,6
4,188,43,52                                           -192-
50010 DATA 57,172,64,188,68,149,64,188,57,172,51,97,6
4,188,51,97,68,149                                    -045-
50020 DATA 57,172,68,149,68,149,57,172,45,198,51,97,5
1,97,45,198                                           -203-
50030 DATA 34,75,43,52,51,97,57,172,61,126,43,52
                                                      -129-
50040 DATA 34,75,43,52,51,97,57,172,61,126,43,52
                                                      -130-
50050 DATA 51,97,43,52,38,126,51,97,38,126,38,126,34,
75,43,53,34,75,34,75                                  -102-
50060 DATA 1,2,11,1,1,1,1,1                           -067-
50070 DATA 240,246,248,249,249,249,249,249            -117-
50080 DATA 238,3,208,206,7,208,238,15,208,238,15,208,
206,9,208,206,9,208                                   -038-
50090 DATA 238,11,208,238,11,208,238,11,208,206,13,20
8,206,13,208,206,13,208,96                            -108-
60000 DATA 206,6,208,206,6,208,206,10,208,206,10,208,
206,12,208,206,12,208                                 -092-
60010 DATA 206,6,208,206,10,208,206,12,208,76,49,192
                                                      -059-
```

TABLON

Os recodarmos nuevamente, que ninguna carta con ánimo de lucro o que huela tan sólo ligeramente a «pirata», olor fétido de descomposición cerebral fácilmente reconocible, será publicada en nuestras páginas.

Además, aprovechamos la ocasión para haceros un ruego: escribid el texto de vuestros anuncios en un papel a parte de cualquier otro que enviéis con sugerencias, etc., y emplead letra de imprenta lo más clara que podáis, a máquina siempre que os sea posible.

ATENCION Commodoreros y spectrumnianos, vamos a crear un club para C-64 y Spectrum con diversos fines: ayudas, intercambios de ideas, concursos, cursillos. etc. Estamos tramitando la edición de una revistilla. Tenemos muchos proyectos, pero sin ti no funcionarán. Para información: Miguel Angel de Rueda. Apdo. Correos 935. 04080 Almería. Llamar de 16,00 a 20,00 h. al Tel.: (951) 34 02 93, preguntar por Toni.

ESTOY INTERESADO en cambio de programas para Vic-20, así como comprar guía de referencia para programador (original o fotocopias). Escribid a: Manuel Moreno Romero. Arzobispo Barrios, 2. 14009 Córdoba.

INTERCAMBIO programas para Commodore 64 en cassette. Rogelio Ferrer Troyano. Avd. Reina Mercedes, 19 A. 41012 Sevilla. Tel.: 61 37 34.

CAMBIO juegos y utilidades para C-64 y C-128 preferiblemente en disco. Contesto a todas las cartas. Tengo más de 700 programas. Carlos Casas Antunez. Doctor Marañón 17-27 6 A. 29009 Málaga.

INTERCAMBIO todo tipo de programas en cinta para CBM-64. Poseo últimas novedades (Nexus, Hardbali, Winter, Games...). Interesados enviar lista a: Alejandro Abadía. Carretera de Daganzo, 8, 6.° B Alcalá de Henares (Madrid).

CAMBIO buenísimos programas por impresora, unidad de disco, joystick o monitor en color que estén en buen estado de uso. Los programas son del tipo Winter Games, Hardball, Rambo, Kawasaky, Rythm Rocker, etc... Cantidad a convenir. Interesados escribir a: José Alberto Criado. Murillo, 3, 1.° B. Alcalá de Henares (Madrid).

CAMBIO programas para C-64 de la categoría de Blackwyche, Summer Games I y II, Zorro... También cambiaría por calculadoras programables, ordenadores de bolsillo. Cantidad a convenir. Interesados escribir a: Andrés Ramos González. Eras de San Isidro, 7, 1.° C. Alcalá de Henares (Madrid).

INTERCAMBIO programas en cinta para Commodore 64. Los interesados escribir a: José María Gómez. Domingo Beltrán, 1-B, 1.° dcha. Vitoria 01008.

CAMBIO SPECTRUM 48K (con manuales, cables, más 300 juegos, interface Kempston, 90 revistas) por unidad de disco para CMB-64. Mi dirección: Mario Castells. Valencia, 17. Tel.: (93) 432 19 43. Barcelona 0815.

DESEO intercambiar programas, tengo unos 180, entre ellos Ghostbusters, Pit StopII, Logo, etc., en cinta, enviar lista a: Andrés Ortega. Fidel Sarel, 13, 1.° izqu., 1.ª Zaragoza 50013.

ME GUSTARIA comprar ordenador C-64 por menos de 25.000 ptas. Mandadme lista. Joaquín Roca i Mallarach. Tel.: 26 03 06. Clot Rosat, 3. Gerona.

ATENCION oferta cambio por Commodore 64 y Datassette en buen estado, emisora de radio aficionado Maxcon CB 40 canales, móvil, fija, antena magnética para automóvil, idem, fija, 40 metros cable blindado antena, fuente alimentación, medidor de estacionarias, manual fácil de radio aficionado. Interesados llamar al Tel.: (93) 421 76 01, o escribir a Antonio Lázaro. Iberia, 9. 0814 Barcelona.

NO SOY «pirata», pero por razones económicas vendo programas que ya no utilizo, asícomo un juego de luces rítmicas (cuatro canales), una emisora de FM (3W) y algunas cosas más (programas de Commodore 64). Tel.: 472 03 99. Madrid. También deseo contactar con programadores amigos del C-64.

VENDO VIC-20 con fuente de alimentación, unidad de ciclo y pantalla, con libros de programación, todo muy nuevo, sin usar (2 semanas); a buen precio. Harald Schillarg. Cuatro, 12. Castelldefels (Barcelona). Tel.: (93) 665 24 33.

¿QUIERES participar en un club de estudiantes de B.U.P. que crea sus propios programas para ayudarse en los estudios? Escribidnos a C.E.B. CO-64. Virgen de los Milagros, 96. Puerto de Santamaría (Cádiz).

TENGO un VIC-20. Quisiera establecer contactos con otras personas que también lo tengan, para intercambiar programas. Pedro Simona. Portal de Cerdana. Seu de Urgen (Lérida).

INTERCAMBIO programas en cinta para Commodore 64. Mandad lista, prometo contestar. Critóbal Dávila Francés. Odena, 36, 3.° Igualada (Barcelona).

ATENCION Commodorianos, desearía intercambiar programas para C-64, juegos y utilidades. Contesto a todas las cartas. Albert Casellas i Tusal. Carme, 9. Igualada (Barcelona).

VENDO VIC-20, fuente de alimentación, dos cartuchos, radar ratace, Sargon II Chess, dos cintas de iniciación, tres libros de programación, varias cintas de juegos, todo por 20.000 ptas. Preguntar pro Manolo. Tenerife, 27, Entresuelo 1. Tel.: 219 72 07. Barcelona 08024.

SI QUEREIS intercambiar programas, experiencias, trucos, proyectos e iniciativas para C-64, escribir al Club Intersoft. Apartado de Correos 27016. Barcelona 08020; o a Manuel Arroyo. Maresma, 266. Barcelona. Entra en el primer Club de intercambio de iniciativas y programas para el C-64.

VENDO VIC-20, ampliación 16 K tarjeta de carga rápida, libros: guía del usuario, guía de referencia del programador, curso de BASIC en dos tomos. Llamar por la noche al Tel.: 254 80 62 de Madrid.

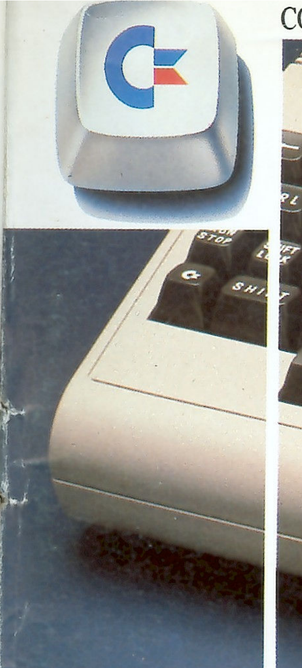
INTERCAMBIO programas para C-64 en cinta; juegos y utilidades (K. Pad Dambusters, Entombed, Superzaxxon,...). Contesto a todas las cartas. Escribir a Juan Fernando Uso Cervera. Juan Bautista Tejedo, 3. Burriana (Castellón).

INTERCAMBIO programas para el CBM-64 por otros de gestión o juegos. Escribid a Jesús Blanco Garcillán. Calasparra, 23. 28033 Madrid.

DESEARIA intercambiar todo tipo de programas para CBM 64. Manda tu lista, prometo contestar. Santi Espinet Mendiola. Estanislao Figueres, 8, 2.° 1.° Tarrassa (Barcelona).

CAMBIO SOFTWARE para Commodore 64 y 128 por ZX-81 o VIC-20 para «Cacharreo» hardware (no importa estado de funcionamiento). Interesados escribir indicando estado de funcionamiento así como lo que desean obtener a cambio, a: José Francisco Aldana, Bethoven, 6, 6.°. 29004 Málaga.

COMMODORE, COMPAÑIA AMERICANA, LIDER MUNDIAL EN NUMERO DE ORDENADORES INSTALADOS.





Más 64's que nadie.

El C-64 de Commodore sigue siendo el ordenador personal más vendido del mundo por sus prestaciones y posibilidades.

Más periféricos que nadie La Gama de periféricos y accesorios del C-64 multiplica sus funciones de una forma casi ilimitada: impresoras, unidades de disco, monitores... Todo un mundo informático a su alrededor para que usted le saque todo el partido.

Más software que nadie El C-64, por ser el ordenador más vendido, ha hecho que las compañías de software se vuelquen en él, creando un parque de programas que hoy le convierten en el 64 con más software del mercado. Software que abarca todos los sectores, desde los negocios hasta el educativo.

Le podemos asegurar que hoy por hoy el programa que usted necesita ya lo tiene el Commodore 64.

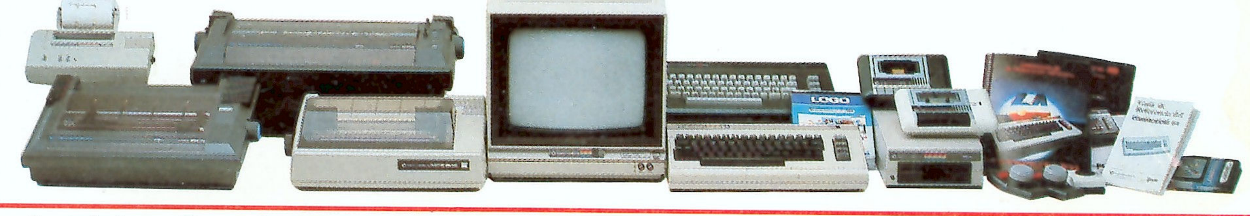
Más información que nadie El C-64, lejos de quedarse atrás y porque cada vez son más los que lo eligen, dispone del mayor número de publicaciones exclusivas, así como libros de documentación en castellano donde se tratan temas de interés, nuevos programas, nuevas ideas, nuevas aplicaciones...

Cada vez más gente investiga y se preocupa por su Commodore 64.

Y menos precio que nadie Y todo esto a un precio realmente sorprendente y asequible.

Acérquese hoy mismo a un distribuidor Commodore y entérese de su precio. Descubrirá qué fácil es poseer el mejor ordenador personal del mundo. El más vendido.





Microelectrónica y Control, s. a. *c/ Valencia, 49-53 08015 Barcelona - c/ Ardemans, 8 28028 Madrid*
Unico representante de Commodore en España.